



Andrew Lacey  
■ El libro gigante de los  
**juegos**  
para **MSX**

**MSX**

# **EL LIBRO GIGANTE DE LOS JUEGOS PARA MSX**

**Andrew Lacey**



**ANAYA MULTIMEDIA**

MICROINFORMATICA

Título de la obra original:  
MSX GAMES BOOK

Traducción de: Celso Losada, Juan Carlos Contreras, Juan José Rodríguez  
Diseño de colección: Antonio Lax  
Diseño de cubierta: Narcís Fernández

Reservados todos los derechos. Ni la totalidad ni parte de este libro puede reproducirse o transmitirse por ningún procedimiento electrónico o mecánico, incluyendo fotocopia, grabación magnética o cualquier almacenamiento de información y sistema de recuperación, sin permiso escrito de Ediciones Anaya Multimedia, S. A.

© 1984 Andrew Lacey

Edición publicada por acuerdo con  
Melbourne House (Publishers) Ltd.,  
Londres

© EDICIONES ANAYA MULTIMEDIA, S. A., 1985  
Villafranca, 22. 28028 Madrid  
Depósito legal: M. 24531-1985  
ISBN: 84-7614-028-2  
Printed in Spain  
Imprime: Anzos, S. A. - Fuenlabrada (Madrid)

# Indice

Introducción.....	7
-------------------	---

## Juegos educativos

El gorila matemático .....	141
Minotauro Mastermind .....	271
Minotauro Mastermind Jr.....	245

## Gráficos interactivos

Diseños .....	21
---------------	----

## Juegos con tiempo límite

Cocodrilo .....	121
Convoy .....	131

## Juego de carreras

Dragster .....	59
----------------	----

## Juegos de disparo

Batalla laser .....	221
Invasión de robots.....	111
O.V.N.I.....	41

## Juegos de evasión

Ataque astral .....	27
Las cucarachas cósmicas.....	67

## Juegos memorísticos

Concentración.....	171
Test de inteligencia .....	75

## Juegos de mesa

En busca del tesoro .....	233
Treinta y una.....	257

## Juegos de simulación

Acróbata aéreo .....	181
Cross .....	201
El saltarín del río .....	83
Emboscada de tanques .....	161
Rescate aéreo .....	151

## Juegos de tiro al blanco

Caseta de tiro .....	211
Globos.....	49
Harrier.....	101
Naves secretas de investigación.....	191
Planetoide.....	33
Prueba de fútbol .....	91

# Introducción

Los ordenadores con el sistema MSX han aparecido para revolucionar el mercado del ordenador personal. Primero ofrecen un BASIC perfeccionado, con posibilidades gráficas y de sonido potentes, haciéndolo un modelo ideal para la programación educativa y de entretenimiento. Segundo, dan a las casas de *software* la ventaja de crear programas ejecutables en una amplia gama de ordenadores; esto implica que el usuario del sistema MSX tiene una amplia gama de programas de primera línea para su ordenador.

Con este libro, Anaya Multimedia continúa con su tradicional edición de libros de alta calidad, tanto literaria como de programas, en beneficio del ordenador personal.

## Organización de los programas

Los programas de juegos están ordenados en este libro, en orden gradual, de más fáciles a más complejos. Sin embargo, y teniendo en cuenta que hay gente que prefiere determinados tipos de juegos, hay un índice que agrupa los programas según el tipo de juego. Si deseas juegos de estrategia, encontrarás los programas consultando el índice.

## Fácil de usar

Para acelerar la depuración de los programas, el programa verificador facilita la detección y corrección en la transcripción de alguna sentencia.

Pero —y esto es importante— primero lee el capítulo «Fiabilidad en los programas», si quieres obtener un máximo beneficio a la hora de transcribir exactamente los programas.

## A la hora de jugar

Para ayudarte a manejar más rápidamente los juegos, se han prefijado las órdenes de juego. Exceptuando los casos que se especifican en la introducción a cada juego, las teclas de espaciado y movimiento de cursor se emplean del siguiente modo:

TECLA	FUNCION
Movimiento del cursor hacia arriba (↑)	Mueve la mira u objeto arriba
Movimiento del cursor hacia abajo (↓)	Mueve la mira u objeto hacia abajo
Movimiento del cursor a la izquierda (←)	Mueve la mira u objeto a la izquierda
Movimiento del cursor a la derecha (→)	Mueve la mira u objeto a la derecha
Barra espaciadora	Botón de disparo

## Objetivos de esta estructuración

Espero que según vayas leyendo y probando los programas de este libro aprendas las bases de cómo programar tu ordenador con el sistema MSX. Para ello hay una lista con las variables, y cada subrutina va encabezada con un título dentro de los listados de cada programa; esto te hará comprender su funcionamiento.

Te desafío a que hagas las mejoras que se te ocurran en los programas de los juegos.

## Sugerencias

Anaya Multimedia estará siempre interesada en recibir correspondencia de sus lectores, ya sean elogios o quejas.

¡Diviértete con los juegos del ordenador!

## FIABILIDAD EN LOS PROGRAMAS

Los programas invierten mucho tiempo en su transcripción y depuración. Para facilitarte la lectura y depuración de éstos, tus listados han sido especialmente estructurados; incluimos el útil programa verificador para facilitarte la ejecución de los programas de juegos. Continúa la lectura de este capítulo e introduce el programa verificador antes que cualquier otro programa de juego.

## FORMATO DE LOS LISTADOS

Los listados de los programas de este libro han sido procesados mediante un programa formateador que los hace más legibles, además de mejorar su presentación. Este programa formateador:

- Alinea las sentencias a la izquierda para evitar que se escalone el margen al ir incrementándose los números de línea.
- Inserta espacios entre las palabras claves.
- Transforma los espacios de las cadenas de caracteres, es decir, entre comillas ("), en un corchete tumbado (↵).

El formateado hace legibles los programas, pero te causará errores de transcripción a menos que sigan las indicaciones que ahora explicamos.

## REGLAS DE TRANSCRIPCION

En los listados de los programas los espacios se han empleado sólo para facilitar su lectura; por ejemplo, la línea:

```
10 PRINT "_CARACTERES_"
```

has de introducirla en tu ordenador como:

```
10 PRINT " CARACTERES "
```

En las cadenas de caracteres se introducen a menudo espacios, pero sólo has de pulsar un espacio cuando veas el símbolo (↵). Este carácter no lo verás en tu teclado; lo incluimos únicamente para que sepas cuándo has de teclear un espacio.

## Caracteres confusos

Existen caracteres muy parecidos que quizá te induzcan a introducirlos incorrectamente y producir un error en tu programa. Estos son: la letra I y el número 1, la letra O y el número 0. En los listados el número cero lo escribimos con una barra atravesándole (Ø). Estos caracteres confusos se reconocen fácilmente en los listados:

CARACTER	LISTADO
Letra I	I
Número 1	1
Letra O	O
Número 0	Ø

Si optas por modificar programas e introducirlos a partir de tus propias notas, te ocurrirá que hay caracteres muy parecidos escritos a mano. Nosotros te aconsejamos:

CARACTER CONFUSO	ESCRITURA RECOMENDADA
Letra Z y número 2	Z, 2
Letra O y número 0	O, Ø
Letra S y número 5	S, 5
Letra I y números 1 y 7	I, 1, 7

Los símbolos de puntuación son también una trampa:

- Las comas (,) y los puntos (.) no son intercambiables.
- Los dos puntos (:) y el punto y coma (;) tampoco lo son.
- Lo mismo sucede con los apóstrofes (') y las comillas (").
- Usa los paréntesis —( )— y los corchetes —[ ]— correctamente. Asegúrate de que están bien emparejados: una forma rápida de comprobarlo es contar los paréntesis o corchetes izquierdos y derechos, y ver que hay la misma cantidad dentro de la expresión matemática.
- La tecla SHIFT has de pulsarla cuando introduzcas algunos caracteres como !, ", %, etcétera.

## Graba (CSAVE) los programas antes de ejecutarlos (RUN)

No te olvides del viejo refrán que dice:

«Los programas transcritos y grabados hoy sobreviven hasta mañana.»

Asegúrate de grabar en *diskette* o cinta los programas según los introduzcas. De vez en vez el teclado se bloqueará, especialmente si tratas de ejecutar un programa con un error. Nosotros te recomendamos, y no lo olvides, que grabes siempre una copia del programa primero y luego elimines todos los errores de transcripción.

**Nota:** Algunos programas tienen ayudas para cargar subrutinas en lenguaje máquina en zonas altas de la memoria y alterarán los punteros del sistema. Si se interrumpe la ejecución de un programa de este tipo, el programa en lenguaje máquina y los punteros quedarán intactos en memoria, incluso si introducimos el comando NEW. Cuando depures los programas, grábalos antes de ejecutarlos e inicializa (RESET) tu ordenador después de un error de ejecución, para evitar continuos errores en la ejecución de los programas transcritos o cargados en memoria.

## COMPROBACION CON EL PROGRAMA VERIFICADOR

Cuando tecleamos los programas de juegos se introducen casi siempre errores de transcripción. Entonces pasamos años intentando ver dónde está la causa del error.

Incluso los programadores expertos no encuentran fácilmente el error y tienen que comprobar con el libro el programa entero, especialmente las sentencias DATA.

Para evitar esta odiosa tarea, ofrecemos dos subrutinas cortas que debes teclear y grabar antes que los programas. Estas son:

- El programa localizador, y
- El programa verificador.

El programa verificador calcula un número para cada sentencia del programa, y lista una tabla con los números de sentencia y su clave, además de una clave total para el programa.

Cuando introduzcas un programa, ejecuta el programa verificador y compara la tabla de claves con la que tienes en el libro. Cualquier diferencia de clave indica un error en la transcripción de dicha línea.

Para que el programa verificador opere correctamente se le ha de dar la dirección de memoria donde comienza el programa en BASIC. El comienzo de la memoria BASIC varía según el tipo de ordenador MSX, teniendo así diferentes mapas de memoria. El localizador es un programa que busca la dirección de comienzo exacta de la memoria BASIC de tu ordenador.

## Grabación del "localizador" y del "verificador"

Sigue estas instrucciones para transcribir y grabar en cinta los programas localizador y verificador. No grabes otros programas en la misma cinta, para no perder tiempo adelantando y rebobinando la cinta al cargar tu programa y al cargar el programa verificador:

1. Introduce el programa localizador que hay un polo más adelante y grábalo en cinta con la orden CSAVE "LOCALI".

**Nota:** Verifícalo carácter a carácter tú mismo, después de haberlo grabado. El localizador es el único programa de este libro que no puede ser comprobado por el verificador. Esto es debido a que para ejecutar una comprobación por el verificador hemos de inicializarlo con la dirección que nos dé el localizador, y si éste está introducido incorrectamente no lo podremos ejecutar.

2. Ejecuta el localizador; mientras se ejecuta aparecerá en pantalla un número en constante incremento; éste se denomina posición. Cuando termine la ejecución aparecerá el siguiente mensaje en pantalla:

Lo encontré; el BASIC comienza en la dirección NNNNNN siendo NNNNN la dirección donde comienzan los programas en BASIC.

Cambia el valor MEBA (memoria de BASIC) en la línea 62000 del programa verificador por este valor.

- Introduce el programa verificador que te damos a continuación. Grábalo en cinta mediante la orden `SAVE"CAS:AUTVER"`; el autoverificador es la versión del programa verificador que se autocorprueba.

**Nota:** La orden `SAVE` hace que el programa verificador se almacene en tu cinta en caracteres ASCII. Esto es necesario, dado que al cargar el programa verificador con la orden `MERGE` a continuación del programa de juego sin borrarlo opera con ficheros de programas en ASCII.

- Carga el verificador para que se autoverifique con la orden `LOAD"CAS:AUTVER"`.
- Ejecuta el programa para que se autoverifique con la orden `RUN`. Se ejecutará sobre si mismo y nos dará una tabla con las claves de sentencia y una clave total para todo el programa.
- En este momento quizá hayas tenido un error o no. Si la tabla ha salido completa incluyendo la clave del programa total, compárala con la tabla que te damos a continuación. Si las claves totales del programa difieren, entonces procede a comparar las claves de cada sentencia con las que te damos; así localizarás el error y lo podrás corregir.

**Nota:** Los programas verificador y autoverificador nos darán a veces diferentes claves para la misma sentencia, incluso estando perfectamente transcrito. Esto se debe a la cantidad de memoria de tu ordenador —mira más adelante las limitaciones del programa verificador en el apartado «Limitaciones».

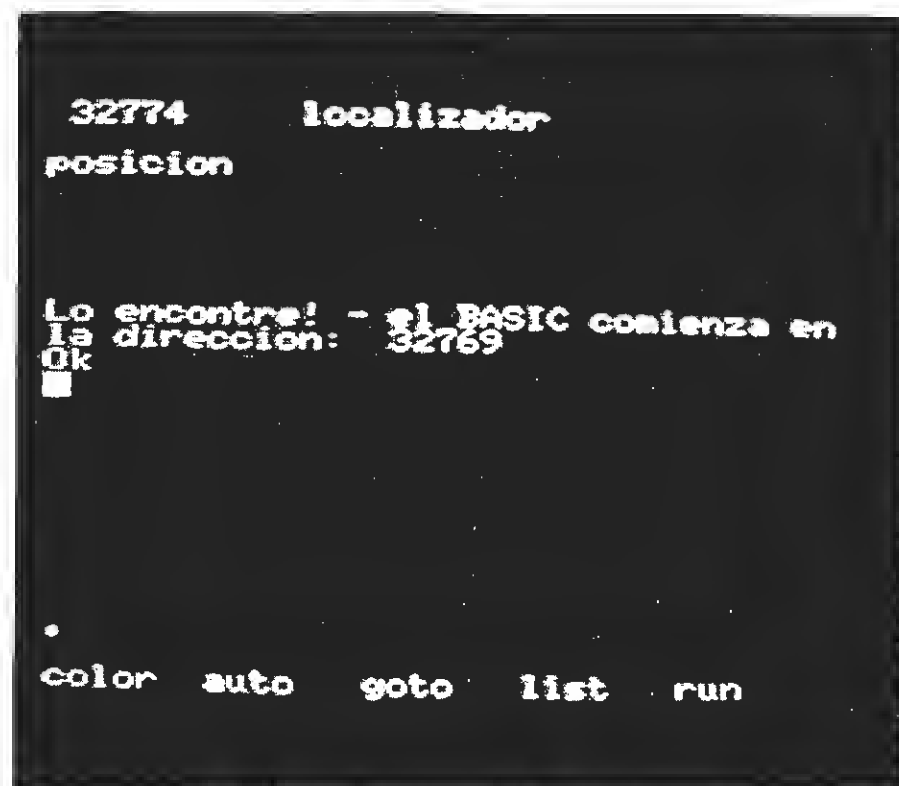
- Repite los pasos del 4 al 7 hasta que tu programa verificador esté correcto.
- En la línea 62020 cambia el valor de `E=62500` por `E=61999`. Graba en tu cinta una copia del programa verificador con la orden `SAVE"CAS:VERIFI"`.

**Nota:** Esta versión, llamada verificador, es tu programa comprobador de los restantes programas del libro. La versión autoverificador sólo se usa para comprobarse a si mismo.

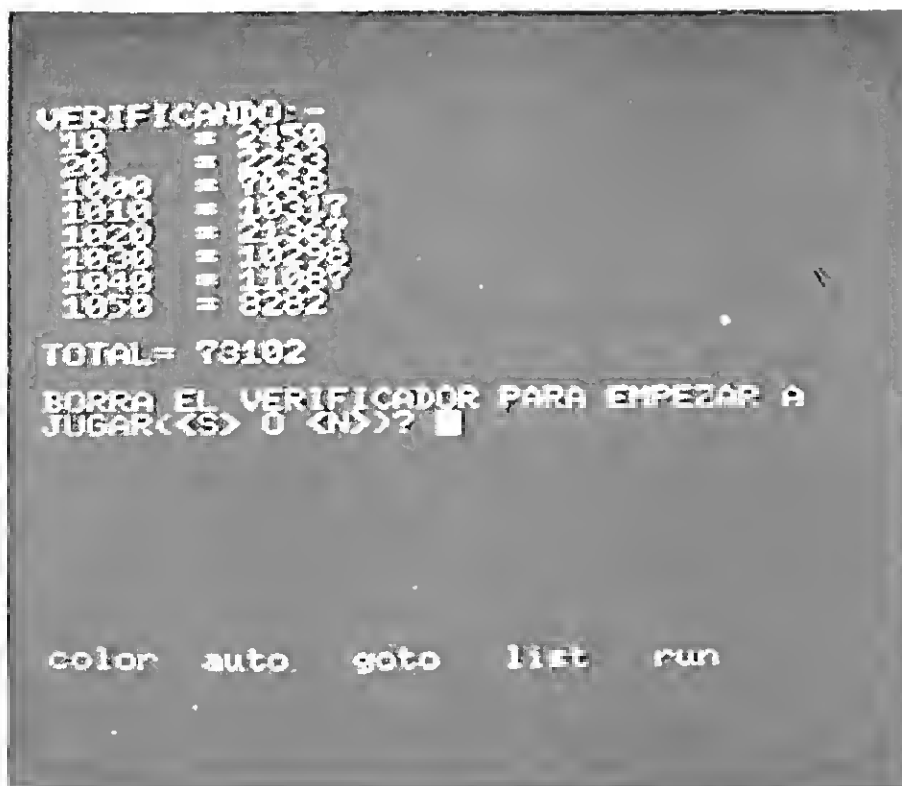
## Programa "localizador"

○	5	REM prueba 12345	○
○	7	B\$=" 32_112_114_117_101_98_97_32_49_50_51":CLS:PRINT: PRINT: PRINT"posicion";CHR\$( 11 )	○
○		; TAB( 11 )"localizador" :PRINT:PRINT: PRINT	○
○	10	PRINT CHR\$( 11 ); MEN:PE=PEEK (MEN): IF PE<>143 THEN MEN=	○
○		MEN+1 : GOTO 10	○
	15	T= MEN : A\$="": FOR I = 1 TO	

○		11 : MEN= MEN+1 : PE = PEEK (	○
	20	MEN ): A\$=A\$+STR\$( PE): NEXT	
○		IF A\$<>B\$ THEN MEN=T+1 :GOTO	○
	30	10	
○		FOR I=1 TO 7: PRINT :NEXT: PR	○
		INT "Lo_encontre!_el_BASIC_	
		comienza_en":PRINT"la_direcci	○
		on:_"; T-5	



## Programa "verificador"



○	62000	MEBA =32769!	○
○	62001	CLS:PRINT TAB(10);"VERIFICADO R":PRINT:PRINT:PRINT	○
○	62004	INPUT "Empiezo en la línea";S T	○
○	62005	CLS:PRINT TAB(10) "VERIFICADO R":FOR I= 1 TO 6:PRINT:NEXT:P RINT TAB(4) "Salida por impre sora" TAB(24) "<I>":PRINT:PRI NT:PRINT TAB(4) "Salida por p antalla" TAB(24) "<P>":FOR I= 1 TO 5:PRINT:NEXT	○
○	62006	PRINT "Teclea 'P' o 'I' "	○
○	62010	X\$=INKEY\$: IF INKEY\$="p" OR X\$	○

○		= "P" THEN 62015 ELSE IF X\$<>" i" AND X\$<>"I" THEN 62010	○
○	62015	REM Rutina impresora	○
○	62020	CLS:PRINT"VERIFICANDO: -":E= 62500!:LINK=MEBA	○
○	62100	REM Bucle principal	○
○	62120	T=LINK	○
○	62130	LINK=PEEK(T+1)*256+PEEK(T)	○
○	62135	LN=PEEK(T+3)*256+PEEK(T+2): IF LN <ST THEN T=LINK:GOTO 62130	○
○	62136	IF LN>E THEN PRINT:PRINT"TOT L=";CH:PRINT:INPUT"BORRA EL V ERIFICADOR PARA EMPEZAR A JUG AR(<S> O <N>)":X\$: IF X\$="S" O R X\$="s" THEN LINK=MEBA :GOTO 62200 ELSE END	○
○	62137	PRINT LN;TAB(7);	○
○	62140	CS=0:N=0:C=0	○
○	62150	FOR P=T+4 TO LINK-2:PK=PEEK(P )	○
○	62160	IF PK=143 THEN P=LINK-2:GOTO 62190	○
○	62165	IF PK=34 THEN C=(C=0)	○
○	62170	IF C=0 AND PK=32 THEN 62190	○
○	62180	IF PK=137 THEN N=N+1:CS=CS+ (203 OR N):PK=164	○
○	62185	N=N+1:CS=CS+(PK OR N)	○
○	62190	NEXT P:CH=CH+CS:PRINT "=";CS:G OTO 62120!	○
○	62200	T=LINK	○
○	62210	LINK=PEEK(T+1)*256+PEEK(T)	○
○	62220	LN=PEEK(T+3)*256+PEEK(T+2)	○
○	62230	IF LN<>62000! THEN 62200 ELSE POKE T,0:POKE T+1,0	○
○	62999	REM	○

El valor MEBA= 32769 lo hemos obtenido de un ordenador de 64K y variará en ordenadores con mayor o menor cantidad de memoria. Mira el apartado «Limitaciones», que hay más adelante.



## Tabla de claves

62000 = 964	62130 = 2583	62185 = 2166
62001 = 2389	62135 = 5488	62190 = 2630
62004 = 2398	62136 = 13621	62200 = 638
62005 = 13306	62137 = 724	62210 = 2583
62006 = 1784	62140 = 1217	62220 = 2698
62010 = 5680	62150 = 2731	62230 = 2982
62015 = 0	62160 = 550	
62020 = 3269	62165 = 1650	
62100 = 0	62170 = 1601	
62120 = 638	62180 = 4133	TOTAL = 78423

## Utilización del verificador

El mayor problema que hay a la hora de transcribir programas de un libro son los errores cometidos al teclearlos. La mayoría se detectan cuando ejecutas el programa con la orden RUN y te aparece un mensaje de error (*Syntax error*). Sólo tienes que listar la línea en cuestión y compararla con la del libro. Desafortunadamente, se cometen errores más sutiles que no son detectados en la ejecución del programa. El programa se ejecuta, pero incorrectamente, y el ordenador no es capaz de detectarlo.

El programa verificador es un programa especial, que genera una clave para cada línea y una clave total del programa. Después del listado de cada programa viene una tabla con los valores de las claves de cada sentencia. Sólo necesitas comparar los números de la tabla de claves de cada programa con los de la tabla que has obtenido. Si dos números son diferentes, comprueba esa línea.

1. Introduce tu programa de juego. Diseños, por ejemplo. Grábalo en cinta con la orden CSAVE"DISENI".

**Nota:** Todos los juegos deben grabarse en cinta mediante la orden CSAVE, que almacena los programas en forma comprimida, dejando más espacio en la cinta y grabando más rápidamente.

2. Si es necesario recargar tu programa en memoria usa la orden CLOAD"DISENI", para el primer programa del libro. En este momento no intentes ejecutar el programa.
3. Para unir el programa verificador al final de tu programa utiliza la orden MERGE"CAS:VERIFI".
4. Cuando lo hayas añadido, teclea RUN 62000 para activar el programa verificador. El programa le preguntará:

¿Número de línea de comienzo?

Pulsando la tecla ENTER hará que en este momento el programa verificador comience en la primera línea de tu programa de juego.

5. Los siguientes mensajes del verificador serán:

Salida a impresora <I>  
Salida a pantalla <P>  
Teclea <I> o <P>

Si tecleas una I, la tabla de salida del verificador la obtendrás por impresora; en cambio, si tecleas una P, la salida la obtendrás por pantalla.

6. El programa verificador te dará la tabla de claves del programa. Para detener la ejecución del verificador presiona la tecla <STOP>, para continuar la ejecución vuelve a presionar <STOP>. Al finalizar la ejecución, el verificador te dirá:

¿Borrado del verificador del programa de juego (S o N)?

Si tecleas una S, el programa verificador se borrará del final de tu programa de juego; por tanto, teclea N.

7. Compara la clave total del programa con las que vienen en el libro. Si son diferentes, es que una línea de tu programa está mal introducida; entonces compara las claves de cada sentencia hasta que localices la o las líneas mal introducidas y corrígelas.

**Nota:** El programa verificador producirá algunas claves erróneas en alguna línea de un programa bien introducido (ver «Limitaciones»).

8. Repite los pasos cuatro a siete hasta que el programa esté correcto. En el apartado 4 introduce el número de la primera línea errónea para permitir que el verificador obtenga la clave después de esa línea y no desde el principio.
9. Cuando el programa ejecute satisfactoriamente, introduce una S en el apartado 6 para borrar el programa verificador del final del programa.
10. Graba en otra cinta tu programa ya depurado usando la instrucción:

CSAVE"DISENI"

## LIMITACIONES

### Verificador

El verificador no puede detectar a veces dos errores, porque se ha obtenido una tabla de claves coincidente con la del libro, pero que realmente no es correcta.

1. Si la clave total coincide con la del libro, pero el programa interrumpe su ejecución por un error, es que dos líneas han sido transcritas incorrectamente; entonces debes comprobar que las claves de todas las sentencias están correctas.
2. Puede ocurrir que en la transcripción hayamos tecleado un punto (.) por una coma (,), o viceversa, y sigan dando una clave correcta para esa línea. Si en una

sentencia DATA se cambia un punto (.) por una coma (,), va a ocurrir que dos números enteros a un lado y otro de la coma se leerán como un único número real y posiblemente genere un mensaje de error en la lectura de un entero, visualizándose en la pantalla "Out of Data". Cuando esto ocurra en cualquier otra sentencia del programa se visualizará el mensaje de error "Syntax error".

Las líneas que tengan GOSUB, GOTO o sentencias que impliquen un salto (por ejemplo, después de una sentencia IF... THEN...) pueden llegar a tener claves diferentes de las del libro, incluso estando perfectamente transcritos los programas.

Este defecto es debido a la cantidad de memoria accesible en el ordenador, y hace que las claves sean diferentes de un ordenador a otro. Todos los programas de este libro han sido probados en un SONY HIT BIT.

Te encontrarás a veces que una línea ha sido perfectamente introducida, y que el programa ejecuta sin ningún fallo; sin embargo, la clave del verificador a esa línea es diferente.

La causa es el modo que tiene de trabajar el sistema MSX; que depende de la cantidad de memoria del ordenador y de las operaciones del editor de líneas. A la hora de editar este libro no nos ha sido posible modificar el programa verificador para eliminar este defecto.

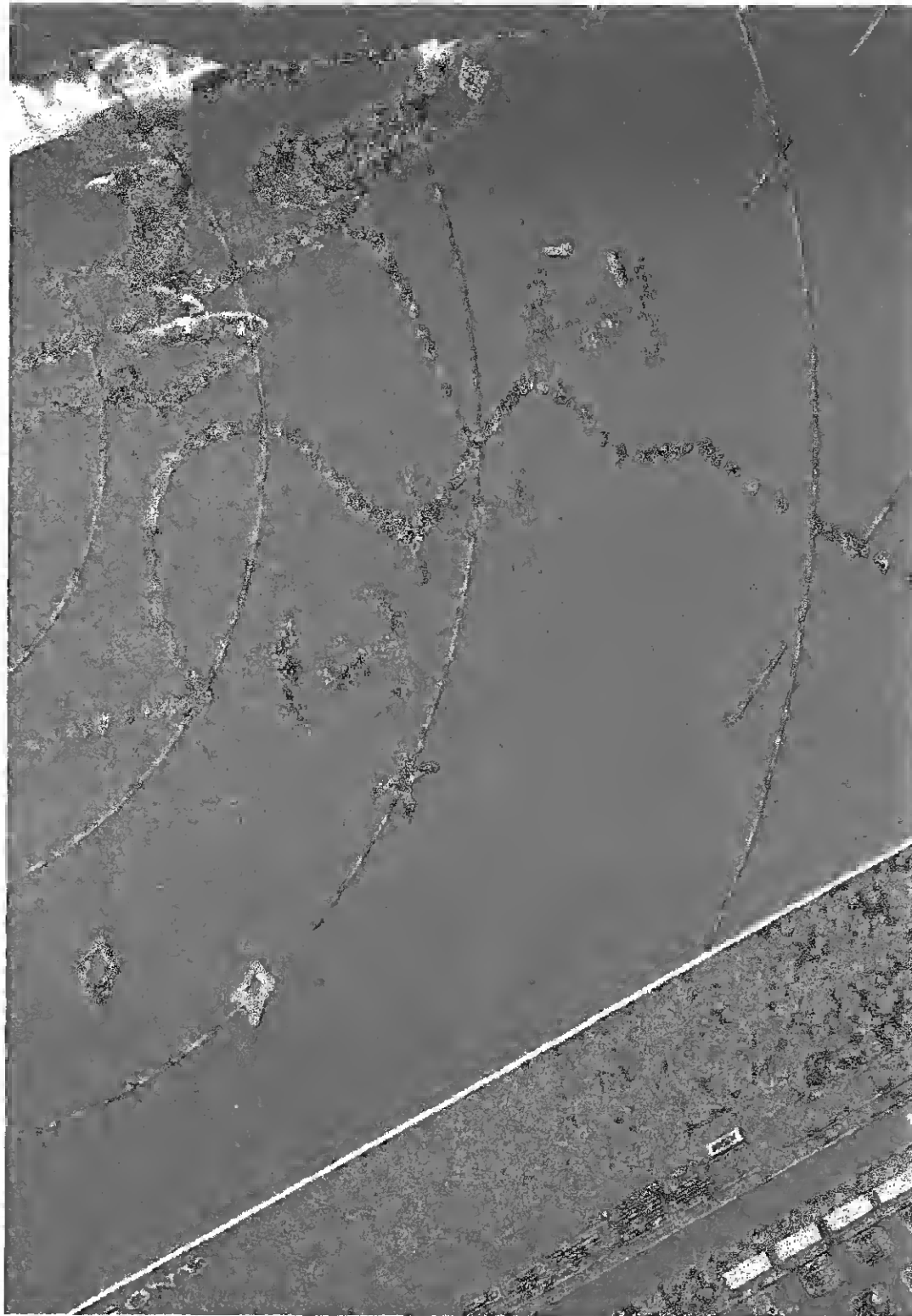
Por tanto, si tu juego está perfectamente introducido y no tiene errores de ejecución, ignora la tabla de claves de verificador.

**Nota:** Es imposible para el verificador eliminar todos los errores de transcripción; el verificador reduce la aparición de estos errores a un nivel mucho más bajo.

## Desajuste de punteros

Hay veces en las que un programa no ejecuta bien, incluso habiendo sido comprobado por el verificador y normalmente; en este caso, hay dos posibilidades:

- Que el programa esté mal —entonces, escribámoslo.
- Una ejecución previa ha dejado mal posicionados los punteros del sistema, haciendo que el programa en curso dé errores de ejecución, incluso empleando la orden NEW. Inicializa siempre el ordenador después de un error en ejecución para evitar ese problema.



# Diseños

O.K.

1



Tipo de juego:

GRAFICOS INTERACTIVOS

¡Crea estupendos diseños con círculos y rectángulos! Este programa te demostrará el superpoder gráfico del sistema MSX. Responde a las preguntas y fíjate en la forma de las figuras.

Los diseños de círculos se forman moviendo los círculos horizontalmente, y cambiando sus formas y medidas.

Los rectángulos se generan a partir de una línea que une dos puntos, P1 y P2.

Pulsa cualquier tecla cuando hayas acabado tu diseño.

## SUGERENCIAS AL PROGRAMA

Aquí tienes algunas combinaciones para que empieces:

CIRCULOS	50,	3,	-0.05,	0
RECTANGULOS	60,	-2,	-2,	-2, -2

Con simples sumas dibujarás líneas en vez de rectángulos o incluso puntos aislados. Puedes hacer cambios: modifica los números de la línea 200 o los de la línea 100.

## Variables del programa

NC	Número de círculos.
IN	Incremento en la dirección del eje X.
VR	Variación del radio.
CR	Cambiar el radio.
CX. CY	Coordenadas del centro del círculo.
R	Radio.
NR	Número de rectángulos.
IX. IY	Incrementos en las coordenadas X e Y del punto P1.
SX. SY	Incrementos en las coordenadas X e Y del punto P2 (han de ser múltiplos de IX e IY).

## Inicialización

```

5      C1=15
10     SCREEN 1 : KEY OFF :PRINT "____
      _____DISEÑOS_____": PR
      INT "_____ ~~~~~~": PR
      INT:PRINT : PRINT
11     PRINT : PRINT TAB(4)"CIRCULOS
      " : PRINT: PRINT :PRINT TAB(4
      )"RECTANGULOS" :PRINT:PRINT:
      PRINT TAB( 4 )"ELIGE_COLOR"
12     FOR I = 1 TO 6 : PRINT : NEXT
      : PRINT "Utiliza las teclas d
      eI_cursor para mover la flech
      a" : PRINT "___PuIsa_'RETURN'_
      para seleccionar"
13     K1 = 60 : GDSUB 19
14     X$ = INKEY$ : IF X$ =CHR$(30)

```

```

THEN IF K=0 THEN 14 ELSE K1=
32 :GOSUB19 :K1 =60: K=K-1 :
GOSUB 19 :GOTO 14
15 IF X$=CHR$(31) THEN IFK=2 THE
N 14 ELSE K1=32:GOSUB19 :K1=
60: K=K+1 :GOSUB 19 :GOTO 14
16 IF X$<>CHR$(13) THEN 14 ELSE
IF K=0 THEN 20 ELSE IF K=1 TH
EN 40 ELSE CLS :INPUT "Tinta_
(1-15)"; X$ :C1=VAL(X$):IF C1
<1 OR C1>15 THEN RUN
17 INPUT "PapeI_(1-15)";X$: K2=V
AL(X$): IF ((K2<1) OR (K2>15))
THEN 17 ELSE INPUT "Borde_(1-
15)";X$
18 K3=VAL(X$): IF K3<1 OR K3 >15
THEN PRINT "Fuera_de_rango":
GOTO 17 ELSE COLOR C1,K2,K3:G
OTO 10
19 VPOKE6356+96*K,K1: VPOKE 6357
+96*K,K1:VPOKE 6358+96*K,K1:R
ETURN
20 CLS : INPUT "Numero_de_circul
os";NC
25 PRINT : INPUT "Incremento_en_
Ia_direccion_deI_eje_X";X$ :
IN = VAL (X$)
30 PRINT : INPUT "Variacion_de_l
a_razon_-_1,1E";X$: IF VAL(X$
)<-1 OR VAL(X$)>1 THEN 30 ELS
E VR = VAL(X$)
35 PRINT : INPUT "Cambiar_eI_rad
io";CR :GOTO 100
40 CLS : INPUT "Numero_de_rectan
gulos";NR
45 PRINT : INPUT "Incremento_en_
Ia_direccion_deI_eje_X";IX :
PRINT :INPUT "Incremento_en_I
a_direccion_del_eje_Y";IY
50 PRINT : INPUT "Incremento_o_d
ecremento_P2_-----coordenada-
X";SX
55 PRINT : INPUT "Incremento_o_d
ecremento_P2_-----coordenada-
Y";SY
90 GOTO 200

```

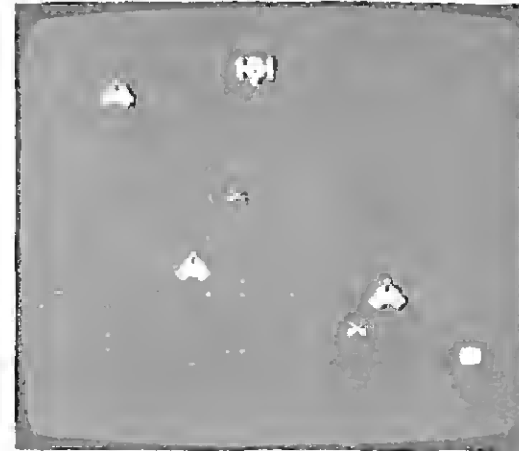
		Círculos	
100	SCREEN 2 : CX = 90 : CY = 90 :		
	IF CR<0 THEN R=100 ELSE IF CR		
	>0 THEN R=5 ELSE R=50		
105	AR = 1		
115	FOR I=1 TO NC		
120	IF CX<0 OR CX>255 OR CY<0 OR		
	CY>192 OR R<=0 OR AR<=0 THEN		
	400 ELSE CIRCLE(CX,CY),R,C1,0		
	,6.28,AR		
125	CX=CX+IN : AR=AR+VR : R=R+CR		
130	NEXT		
190	GOTO 400		
		Rectángulos	
200	SCREEN 2 : X1=100 : Y1=100 : X2		
	=150 : Y2=90		
210	FOR I=1 TO NR : X1=X1+IX : Y1		
	=Y1+IY : X2=X2+5X*IX : Y2=Y2+S		
	Y*IY		
220	LINE (X1,Y1)-(X2,Y2),C1,B		
230	NEXT		
240	GOTO 400		
		Espera	
400	IF INKEY#="" THEN 400		
410	GOTO 10		

## Tabla de claves

4	= 337	25	= 5647	130	= 131
5	= 387	30	= 7234	190	= 537
10	= 5541	35	= 2956	200	= 2582
11	= 6057	40	= 2987	210	= 6152
12	= 11076	45	= 9412	220	= 1645
13	= 697	50	= 4985	230	= 131
14	= 6340	55	= 5019	240	= 537
15	= 5494	90	= 593	400	= 1091
16	= 9208	100	= 4971	410	= 403
17	= 6659	105	= 408		
18	= 6648	115	= 837	TOTAL= 134745	
19	= 4582	120	= 7932		
20	= 2599	125	= 2930		

# Ataque astral

# 2



Tipo de juego:

JUEGO DE EVASION

Has conseguido llevar a tu enemigo hasta un vórtice, donde no funciona ningún tipo de arma. Choca contra las naves pequeñas, pero ten cuidado con las grandes.

La batalla termina cuando se te escapa una de las naves pequeñas.

Utiliza las teclas de movimiento de cursor para moverte a la izquierda y a la derecha.

## SUGERENCIAS AL PROGRAMA

Intenta un efecto espectacular de explosión; cuando haya un choque, hazlo cambiando las líneas 800 a 830.

La línea 200 controla el espacio vertical entre las naves enemigas y la 215 controla la distancia horizontal; modificándolas podrás tener una formación de naves enemigas diferente.

## Variables del programa

NP Número de naves pequeñas destruidas.  
DN Distancia para mover todas las naves.  
SA Aparición de un sprite en pantalla.  
VS Ultimo sprite en pantalla.  
NE Sprite que está enfrente (nave enemiga).  
FJ Final de juego.  
DI Nivel de dificultad.

		Inicialización	
5	SCREEN 1,2 : KEY OFF : COLOR 15,1,1 : PRINT "*****ATAQUE LASTRAL"		
10	REM EJECUTA EL PROGRAMA		
11	REM SOPORTE EN CODIGO		
12	REM MAQUINA		
13	REM VER APENDICE		
15	FOR I = 1 TO 32: READ Q:A\$=A\$+CHR\$(Q): NEXT: SPRITE\$(0)=A\$:A\$=""		
20	FOR I=1 TO 32: READ Q: A\$=A\$+CHR\$(Q): NEXT: SPRITE\$(1)=A\$:A\$=""		
25	FOR I = 1 TO 8: READ Q:A\$=A\$+CHR\$(Q):NEXT: SPRITE\$(2)=A\$		
30	DEFUSR=60000!: DEFUSR1=60118!:POKE 59996!,10		
35	FOR I = 1 TO 5: VPOKE 6914+4*I,8:NEXT		

40	NE=1:PRINT:PRINT:PRINT: INPUT "NIVEL_DE_DIFICULTAD_(1-4)"; DI\$:DN=2*VAL(DI\$)+2: IF DN>10 THEN 40		
50	FOR I = 1 TO 8: PRINT: NEXT: PRINT "PULSA UNA TECLA PARA EMPEZAR"		
55	D=RND(1): IF INKEY\$="" THEN 55		
60	SPRITE DN:DN SPRITE GOSUB 800		
65	FOR I = 1089 TO 1095: VPOKE I,0: NEXT: VPOKE 1088,1		
70	CLS: FOR I = 1 TO 60 : VPOKE 6144+INT(RND(1)*700),136:NEXT		
95	TIME=0:PUT SPRITE 0,(120,160),11		
	Control		
100	POKE 59999!,7:D=USR(D)		
110	GOSUB 200		
120	GOSUB 300		
190	GOTO 100		
	Nave nueva		
200	IF TIME<105-25*DI THEN RETURN		
210	K=US+1:IF K= 6 THEN K=1		
212	IF SA AND 2^K THEN RETURN ELSE US=K:TIME=0:IF RND(1)<.6 THEN K1=8: K2=7 ELSE K1=4:K2=8		
215	VPOKE 6914+4*US,K1:PUT SPRITE US,(INT(RND(1)*(150-20*DI)+40+20*DI),0),K2:SA=(SA) OR 2^US		
220	RETURN		
	Movimiento de naves		
300	POKE 59999!,DN:POKE 59998!,2: FOR I=1 TO 5:IF SA AND 2^I THEN POKE 59997!,I:D=USR1(D):IF		

```

301 VPEEK(6912+4*I)>190 AND(SA AND
    D 2^I) THEN SA=SA AND(255-2^I
    ):PUT SPRITE 1,(10+30*I,200)
    IF VPEEK(6914+4*I)=8 THEN FJ=
    1:GOTO 900 ELSE NE=NE+1:IF NE
    =6 THEN NE=1
305 NEXT
310 RETURN

```

Choque

```

800 SPRITE OFF: IF VPEEK (6914+4*
    NE)=4 THEN 900
810 PUT SPRITE NE,(10+30*NE,200):
    SA=SA AND (255-2^NE):ND=ND+1:
    PLAY "154m1700s10n54n32"
820 NE=NE+1:IF NE=6 THEN NE=1
830 SPRITE ON: RETURN

```

Fin de juego

```

900 SCREEN 1: PRINT "____FIN
    _DE_JUEGO":PRINT :PRINT:PRINT
    "____PARA_OTRA_PARTIDA":PR
    INT"____INSERTA_MONEDA"
910 FOR I=1 TO 15:PRINT:NEXT : PR
    INT "____NAVES_DERRIBADAS";
    ND:CHR$(11);
920 FOR I=1 TO 7: PRINT: NEXT: IF
    FJ=1 THEN PRINT: PRINT "____
    NAVE_ENEMIGA_ESCAPADA" ELSE P
    RINT "____DESTRUIDO_POR_LU
    NA":PRINT: PRINT "____NAVE_
    DE_TRANSPORTE!"
940 IF INKEY$="" THEN END ELSE
    940

```

Datos de sprites

```

10000 DATA 1,1,1,129,129,199,206,
    206,254,206,255,207,199,198,

```

```

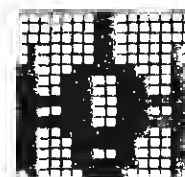
199,131,128,128,128,130,130,
227,115,115,127,115,255,243,
227,99,227,194
10010 DATA 56,56,255,255,255,127,63
    ,63,31,30,14,6,2,1,1,1,28,28,
    255,255,255,254,252,252,248,
    120,112,96,64,128,128,128
10020 DATA 129,195,231,255,24,36,66
    ,129

```

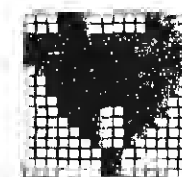
## Tabla de claves

5	= 3146	70	= 3625	800	= 2610
10	= 0	95	= 1549	810	= 6496
11	= 0	100	= 1493	820	= 2089
12	= 0	110	= 362	830	= 551
13	= 0	120	= 207	900	= 7205
15	= 3995	190	= 489	910	= 4256
20	= 3996	200	= 1764	920	= 11259
25	= 3500	210	= 1873	940	= 1481
30	= 2746	212	= 6354	10000	= 10608
35	= 1951	215	= 7584	10010	= 8870
40	= 6918	220	= 143	10020	= 1733
50	= 4004	300	= 14070		
55	= 1893	301	= 5631	TOTAL=	137664
60	= 926	305	= 131		
65	= 2013	310	= 143		

## Diseño de los sprites



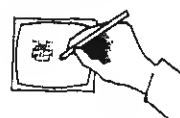
NAVE PROPIA



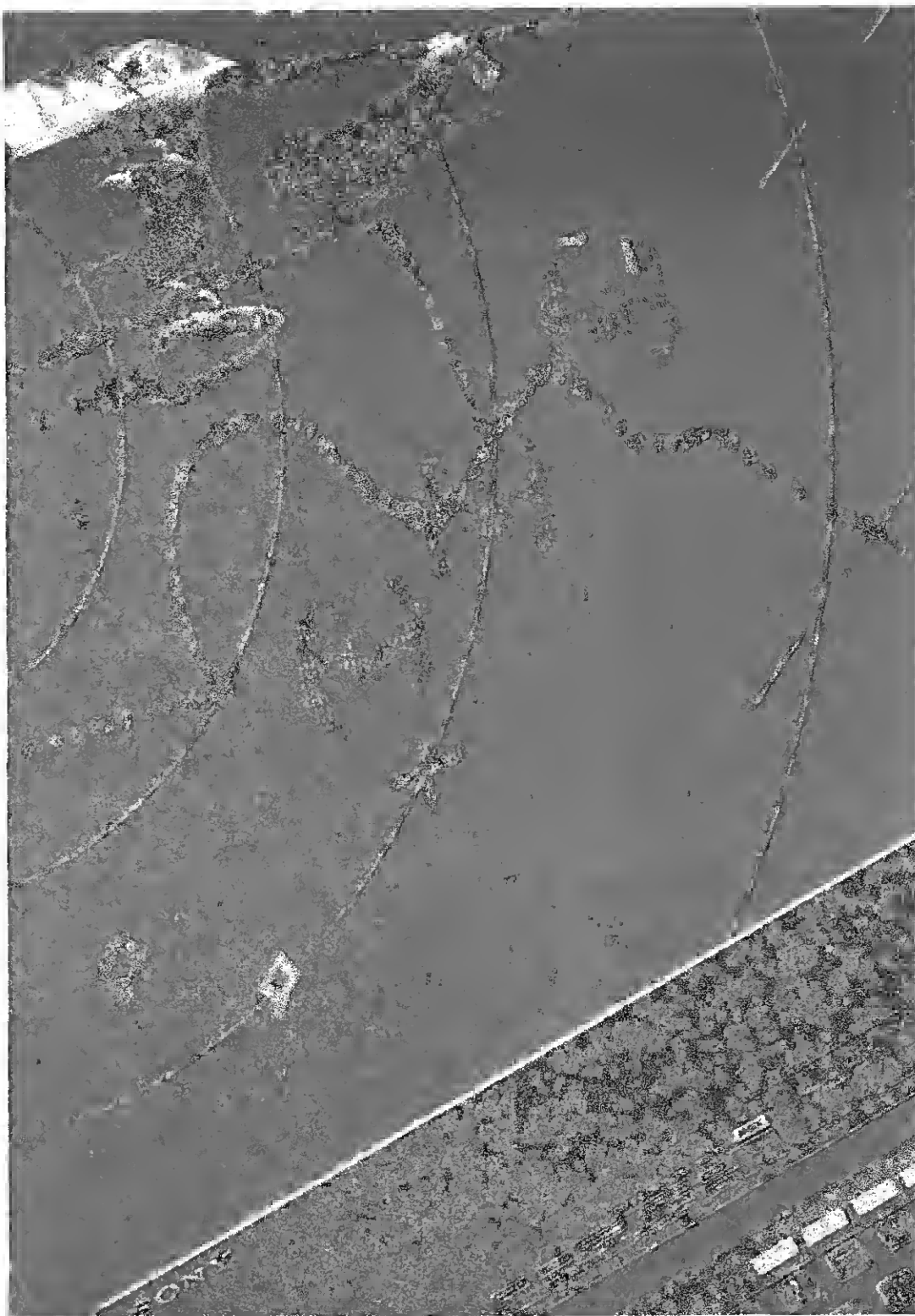
TRANSPORTE



CAZA

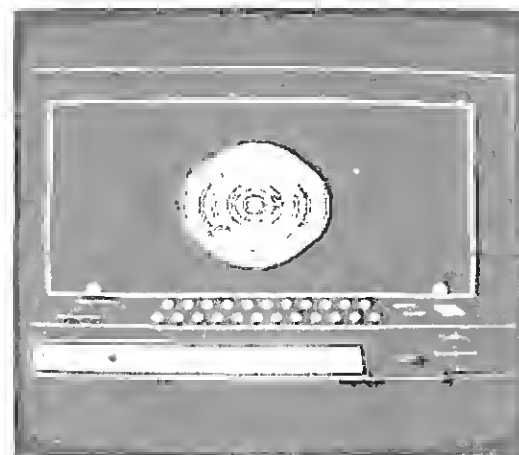






# Planetoide

# 3



Tipo de juego:

JUEGO DE TIRO AL BLANCO

Atrévete a atravesar un corredor espacial lleno de planetas destructores. Tus lasers deben incidir en un pequeño punto escondido en el planeta, a no ser que quieras convertirte en polvo cósmico.

Observa el ambiente que crean los espectaculares gráficos en esta rara zona del espacio.

## SUGERENCIAS AL PROGRAMA

La proximidad del planeta está controlada por las líneas del programa 450 a 490; sería muy interesante que hicieras alguna modificación.

Haz que R se incremente más rápidamente, o cambia el valor de VR para obtener un aspecto diferente del planeta.

También puedes cambiar esta zona del programa para complementar diversos niveles de dificultad.

### Variables del programa

PL	¿Está el planeta en pantalla?
R	Radio del planeta.
PX, PY	Coordenadas del punto a destruir.
CX, CY	Centro del planeta.
VR	Variación del radio del planeta.
NP	Número de planetas destruidos.

○			○
○		Inicialización	○
○	5	CLS:PRINT "_____PULSA_U NA_TECLA"	○
○	6	IF INKEY\$="" THEN D=RND(1):GO TO 2	○
○	7	COLOR 15,1,1:SCREEN 2,2	○
○	10	REM EJECUTA EL PROGRAMA	○
○	11	REM SOPORTE EN CODIGO	○
○	12	REM MAQUINA	○
○	13	REM VER APENDICES	○
○	15	FOR I=1 TO 32: READ Q: A\$=A\$+ CHR\$(Q): NEXT: SPRITE\$(0) = A\$	○
○	20	DEFUSR=60000!:POKE 59996!,15: POKE 59999!,8	○
○	30	A\$="":FOR I=1 TO 8:READ Q: A\$= A\$+CHR\$(Q): NEXT: SPRITE\$(1)= A\$	○
○		Preparación de pantalla	○
○	100	LINE (20,140)-(235,20),15,B:L INE(0,160)-(255,0),15,B: LINE	○

○		(0,192)-(255,160),15,B: LINE (10,190)-(180,170),15,B	○
○	110	LINE (13,187)-(177,173),15,B: PAINT (50,186)	○
○	120	LINE(0,160)-(20,140):LINE(255 ,160)-(235,140):LINE(0,0)-(20 ,20):LINE (255,0)-(235,20)	○
○	130	FOR I=80 TO 180 STEP 10: CIRC LE(I,146),2: PAINT (I,146): N EXT: FOR I= 75 TO 185 STEP 10 : CIRCLE(I,154),2: PAINT (I , 154):NEXT	○
○	140	DRAW "bm35,145r20g10120e10bm1 95,147r10f5110h5bm215,147r10f 5110h5":PAINT (220,150)	○
○	150	DRAW "bm200,175g5r10h5":CIRCL E(225,177),10:LINE(225,167)-( 225,187): LINE(215,177)-(235, 177)	○
○	160	LINE(21,141)-(234,21),15,B:DR AW "bm40,139u4e3f3d4bm215,139 u4e3f3d4":PAINT (42,138):PAIN T (217,138)	○
○	165	IF NP=0 THEN 170 ELSE DRAW "b m35,8":FOR I=1 TO NP: DRAW "g 3f3e3h3br9": NEXT	○
○	170	PRESET(17,178): PRESET (17 , 179): PRESET (17,180): PRESET (18,179)	○
○	190	IF NP <> 0 THEN 270	○
○	200	FOR J= 1 TO 3: X1=120:X2=141: Y1=80:Y2=90	○
○	210	FOR I=1 TO 10:B\$="n"+STR\$(I+ 20): PLAY "m299s1119xb\$":LIN E(X1,Y1)-(X2,Y2),15,B:X1=X1-I :X2=X2+I:Y1=Y1-I:Y2=Y2+I: NEX T: LINE (45,130)-(211,24),15, B	○
○	220	X1=120:X2=141:Y1=80:Y2=90:FOR I=1 TO 10:B\$="n"+STR\$(30-I):P LAY"m299s1119xb\$":LINE(X1,Y1 )-(X2,Y2),1,B:X1=X1-I:X2=X2+I :Y1=Y1-I:Y2=Y2+I:NEXT:LINE(45 ,130)-(211,24),1,B:NEXT	○
○	270	PUT SPRITE 0,(120,70),9	○

```
280 PX=0:PY=0:STRIG(0)ON:ON STRIG
GOSUB 900
```

Control

```
300 A=PEEK(59996!):X=VPEEK(6913):
Y=VPEEK(6912):IF X<30 THEN P
OKE 59996!,PEEK(59996!) AND
13 ELSE POKE 59996!,PEEK(
59996!) OR 2
305 IF X>209 THEN POKE59996!,PEEK
(59996!) AND 7 ELSE POKE
59996!,PEEK(59996!) OR 8
310 IF Y<30 THEN POKE59996!,PEEK(
59996!) AND 14 ELSE POKE
59996!,PEEK(59996!) OR 1
315 IF Y>100 THEN POKE59996!,PEEK
(59996!) AND 11 ELSE POKE
59996!,PEEK(59996!) OR 4
320 D=USR(D)
330 IF PL=0 AND RND(1)<.1 THEN GO
SUB 400
340 IF PL=1 THEN GOSUB 450
390 GOTO 300
```

Aparición planetoide

```
400 CX=INT(RND(1)*90+80):CY=INT(R
ND(1)*30+60):R=3:CIRCLE(CX,CY
),R:PAINT(CX,CY)
410 PL=1:PUT SPRITE 1,(170,176),1
420 PX=CX+INT(RND(1)*80-40):PY=CY
+INT(RND(1)*70-35)
440 RETURN
```

Agrandamiento de planetoide

```
450 IF RND(1)<.2 THEN R=R+2
460 VR=RND(1)/3+.8:CIRCLE(CX,CY),
R,15,0,6.28,VR
```

```
470 VPOKE6917,160-3*R
480 IF VPEEK(6917)<15 THEN 2000
490 RETURN
```

Planetoide destruido

```
700 STRIG(0) OFF:FOR I=1 TO 50 :P
LAY"140m380s8n24":X1=INT(RND(
1)*90-45+CX):Y1=INT(RND(1)*90
-45+CY):X2=INT(RND(1)*200+25)
:Y2=INT(RND(1)*110+25)
710 LINE(X1,Y1)-(X2,Y2),1:CIRCLE(
X1,Y1),1:CIRCLE(X2,Y2),2:PAIN
T(X2,Y2):NEXT
720 PUT SPRITE 1,(100,200):PL=0
730 NP=NP+1:CLS:GOTO 100
```

Disparo del laser

```
900 IX=VPEEK(6913)+8:IY=VPEEK(
6912)+8:LINE(44,132)-(IX,IY),
11:LINE(217,132)-(IX,IY),11
905 PLAY"110m1000s14n33"
910 LINE(44,132)-(IX,IY),1:LINE(
217,132)-(IX,IY),1
920 IF (PX-CX)^2+(PY-CY)^2>R*R T
HEN RETURN
930 IF ABS(IX-PX)<15 AND ABS(IY-P
Y)<15 THEN 700
970 RETURN
```

Fin de juego

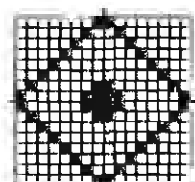
```
2000 FOR I=1 TO 100 : PLAY "164m20
0s14n23":X1=INT(RND(1)*255):Y
1=INT(RND(1)*192):X2=INT(RND(
1)*255):Y2=INT(RND(1)*192):LI
NE(X1,Y1)-(X2,Y2):NEXT
2010 SCREEN 1:PRINT "-COLISION_CO
N_EL_PLANETOIDE!":PRINT:PRINT
:PRINT:PRINT"PLANETOIDES_DEST
RUIDOS: ";NP
```

2099	IF INKEY\$="" THEN END ELSE
2099	
	Datos de sprites
10000	DATA 1,2,4,8,16,32,65,131,131,65,32,16,8,4,2,1,128,64,32,16,8,4,130,193,193,130,4,8,16,32,64,128
10010	DATA 24,60,126,126,60,24,0,0

## Tabla de claves

1	= 1997	200	= 3128	480	= 1454
2	= 2253	210	= 13684	490	= 143
5	= 723	220	= 18153	700	= 15021
10	= 0	270	= 883	710	= 4038
11	= 0	280	= 2712	720	= 1397
12	= 0	300	= 9391	730	= 1623
13	= 0	305	= 5207	900	= 6442
15	= 3482	310	= 5098	905	= 1301
20	= 2455	315	= 5158	910	= 3119
30	= 4010	320	= 691	920	= 3645
100	= 8141	330	= 2309	930	= 3657
110	= 2342	340	= 1155	970	= 143
120	= 5656	390	= 433	2000	= 14040
130	= 6409	400	= 6874	2010	= 6916
140	= 5772	410	= 1554	2099	= 1362
150	= 6368	420	= 5282	10000	= 7039
160	= 6637	440	= 143	10010	= 1398
165	= 4450	450	= 1928		
170	= 2646	460	= 3705	TOTAL=	225751
190	= 1069	470	= 1115		

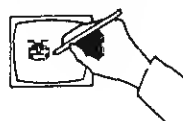
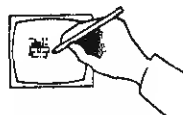
## Diseño de los sprites

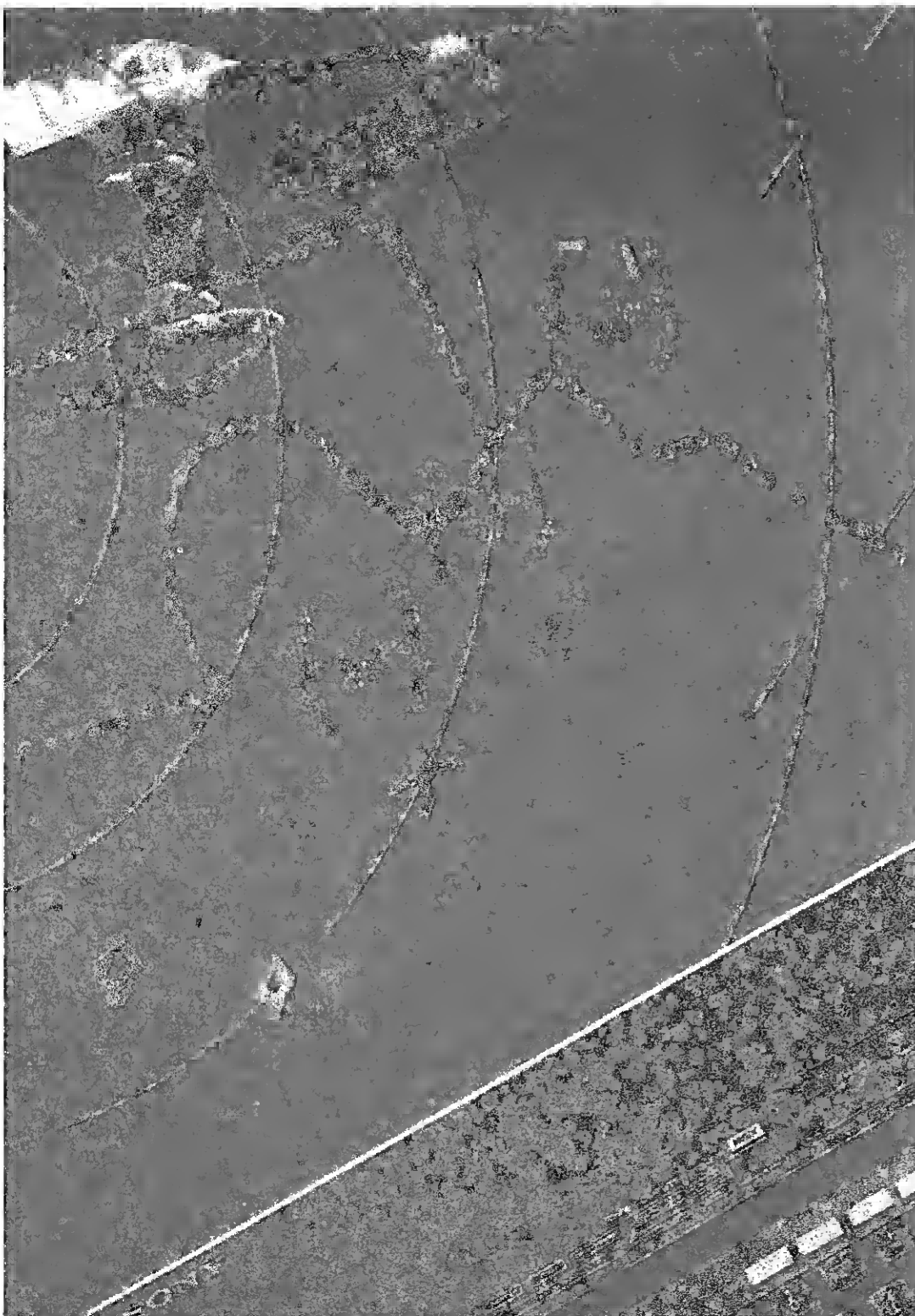


PUNTO DE MIRA



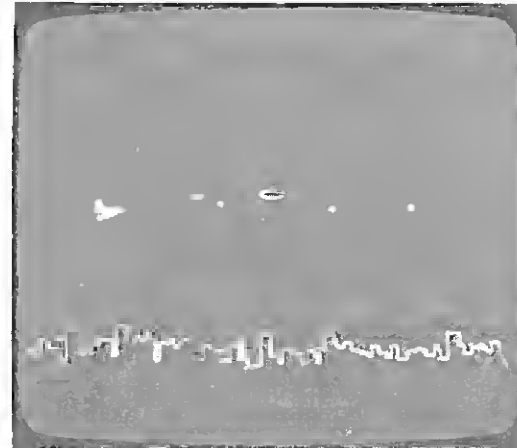
PLANETOIDE  
EN LA PANTALLA DE RADAR





# O.V.N.I.

# 4



Tipo de juego:

**JUEGO DE DISPARO**

Da caza a los O.V.N.I.s en tu nave espacial equipada con misiles. Lograrás despegar de la Tierra si no chocas con el terreno o con un O.V.N.I. Date prisa, tu tiempo se va agotando.

Utiliza las teclas de movimiento de cursor para subir, bajar y variar tu velocidad; para disparar, teclea la barra espaciadora. Para que aparezca tu nave al comienzo, pulsa el cursor hacia abajo.

## SUGERENCIAS AL PROGRAMA

Fija el tiempo de duración de cada pantalla; T1 y T2 se ajustan en la línea 60.

Si te sientes con ganas, añade otra nueva pantalla a continuación de la segunda; pero la tienes que diseñar tú mismo.

## Variables del programa

DM Disparo de misil.  
T1, T2 Límites de tiempo de las pantallas 1 y 2.  
NO Número de O.V.N.I.s destruidos.  
FJ Cómo finalizó el juego.

		Inicialización	
10	REM	EJECUTA EL PROGRAMA	
11	REM	SOPORTE EN CODIGO	
12	REM	MAQUINA	
13	REM	VER APENDICES	
15	COLOR	15,1,1: SCREEN 2,2	
20	FOR I= 1 TO 32 : READ Q : A\$=		
	A\$ + CHR\$ (Q) : NEXT : SPRITE		
	\$ (0) = A\$ : A\$=""		
25	FOR I= 1 TO 32 : READ Q : A\$		
	= A\$ + CHR\$ (Q) : NEXT : SPRI		
	TE\$ (1) = A\$ : SPRITE\$ (2)=		
	CHR\$ (255) + CHR\$ (255) : FOR		
	I = 14432 TO 14464: VPOKE I,2		
	^ (INT(RND(1)*8))		
27	NEXT		
30	DRAW "BM10,30D134R1D4R2D3R3D2		
	R4D1R55U1R4U2R3U3R2U4R1U134L2		
	1D110L1D4L1D3L1D2L1D1L25U1L1U		
	2L1U3L1U4L1U110L20"		
35	DRAW "BM99,30D145R20U60R30U20		
	L30U45R55U20L75": PAINT (100,		
	100)		
40	DRAW"BM195,30R40D1R4D2R3D3R2D		
	4R1D126L1D4L2D3L3D2L4D1L40U1L		
	4U2L3U3L2U4L1U126R1U4R2U3R3U2		
	R4U1"		

45	DRAW "BM208,51R14D1RD1R2D2R1D		
	98L1D2L2D1L14U1L2U2L1U98R1U2R		
	2U1": PAINT (200,60)		
50	PUT SPRITE 1, (0,5),8:DEFUSR		
	= 60000!: DEFUSR1 =60118!: PO		
	KE 59996!,13 : POKE 59997!,1		
	: POKE 59998!,3 : POKE 59999!		
	,1		
55	IF INKEY\$="" THEN D=USR1(D) :		
	GOTO 55		
60	T1=60*60: T2= 69*30 : TIME=0		
		Pantalla 1	
100	CLS : COLOR 15,1,1 : DRAW "bm		
	0,170r5d12r2u3r6d4r2u10r3d2r2		
	d3r4u6r2d14r2u20r7d15r3u2r2d2		
	r1u6r4u3r7d9r2u10r5d10r3u1r2u		
	3r2d2r3u4r2d3r3d14r2u11r6u2r2		
	d5r2u2r2u3r3d7"		
105	DRAW "r3d5r5u8r3d1r3d2r3d6r5u		
	9r4d8r2u12r1d12r2d5r4u1r3u19r		
	5d13r4d4r2u8r4d2r3d1r3d5r3u7r		
	3d2r2d5r1u8r3u9r5d3r1d4r4u2r4		
	d4r3d1r2d1r3d2r2u6r3d2r1d2r4d		
	3r2u7r3"		
110	DRAW "d8r5u3r1u2r4u1r3d1r2d3r		
	4u6r2d3r1d5r4u16r6d8r2d5r4u3r		
	4u2r2d3r2d2r2u6r4d6m255,192"		
120	PUT SPRITE 1, (120,70),8: PUT		
	SPRITE 2, (0,80),11		
130	STRIG(0) ON: ON STRIG GOSUB		
	400		
140	SPRITE ON : ON SPRITE GOSUB		
	450		
		Control 1	
200	IF VPEEK(6912)>175 THEN POKE		
	59996!,12 ELSE POKE 59996!,13		
205	D= USR (D) : IF VPEEK(6912)>		
	160 AND VPEEK (6912)<175 THEN		

```

210 CJ=2: GOTO 800
    POKE 59998!,3 : POKE 59999!,4
    : POKE 59997!,0 : D =USR1(D)
220 GOSUB 300
230 IF DM =1 THEN GOSUB 350
240 IF TIME>T1 THEN GOTO 500
290 GOTO 200

```

Movimiento O.V.N.I.

```

300 POKE 59997!,1: POKE 59999!,7
    : IF DM=0 THEN POKE 59998!,3
    : GOTO 310
305 IF RND (1)<.6 THEN POKE
    59998!,3 ELSE IF RND(1) <.52
    THEN POKE 59998!,0 ELSE POKE
    59998!,2
310 D=USR1(D)
320 RETURN

```

Movimiento del avión

```

350 IF VPEEK(6921)>238 THEN DM=0:
    PUT SPRITE 2, (101,200):RETURN
360 VPOKE 6921,VPEEK(6921)+16
370 RETURN

```

Disparo del avión

```

400 IF DM=1 OR VPEEK(6913)>243
    THEN RETURN
405 DM=1 : PUT SPRITE 2, (VPEEK(
    6913)+16,VPEEK(6912))
410 RETURN

```

Choque

```

450 SPRITE OFF : IF DM =0 AND ABS
    (VPEEK(6913)-VPEEK(6917))< 17
    THEN CJ= 1 : VPOKE 6914,12 :

```

```

455 FOR I1 =1TO 200 : NEXT : VPOK
    E 6918,12: FORI1 = 1 TO 800 :
    NEXT : GOTO 800
    IF ABS(VPEEK(6916)-VPEEK(6920
    ))<8 THEN NO=NO+1:VPOKE 6918,
    12:PUT SPRITE 2, (101,200):PLA
    Y "14m2000s14n15": FOR I1=1TO
    500: NEXT: VPOKE 6918,4:DM=0:
    PUT SPRITE 1, (0,70+INT(RND(1)
    *20)):SPRITE ON:RETURN
460 DM = 0 : PUT SPRITE 2, (101,
    220): SPRITE ON : RETURN

```

Pantalla 2

```

500 COLOR 15,1,1: CLS
505 CIRCLE(220,5),5,15,0,6,.5
510 FOR I=1 TO 9: CIRCLE(220,5+3*
    I),10+2*I,15,0,6,28,.5: NEXT
    : FOR I=1 TO 9 : CIRCLE(220,
    33+3*I),28-2*I,15,0,6,28,.5 :
    NEXT
590 FOR I=1 TO 255 STEP 6 : FOR J
    =0 TO 190 STEP 6 : IF RND(1)<
    .1 THEN PSET(I,J),15
595 NEXT : NEXT :TIME =0: SPRITE
    ON

```

Control 2

```

600 D=USR(D)
610 POKE 59998!,3 : POKE 59999!,5
    : POKE 59997!,0: D=USR1(D)
620 GOSUB 300
630 IF DM=1THEN GOSUB 350
640 IF TIME>T2 THEN CJ=3: GOTO
    800
690 GOTO 600

```

Fin de juego

```

800 SCREEN 1 : PRINT ".....

```

```

805  _FIN_DE_JUEGO": PRINT : PRINT
PRINT "HAS_DESTRUIDO_";NO;"_LO
.V.N.I._INVASORES": PRINT :
PRINT
810  IF CJ=1 THEN PRINT "COLISION_
CON_UN_O.V.N.I."
820  IF CJ=2 THEN PRINT "CHOCASTE_
CON_LA_SUPERFICIE!"
830  IF CJ=3 THEN PRINT "TIEMPO_AC
ABADO: INVASORES": PRINT "REGRE
SAN_A_SUL_PLANETA_DE_...ORIGE
N"
890  IF INKEY$="" THEN END ELSE
890

```

Datos de sprites

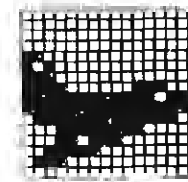
```

10000 DATA 0,0,0,224,224,224,240,
248,255,255,127,31,31,62,112,
0,0,0,0,0,0,0,56,254,255,
252,224,128,0,0,0
10010 DATA 0,0,0,3,5,15,126,255,240
,127,63,7,0,0,0,0,0,0,192,
64,240,102,255,0,254,252,224,
0,0,0,0

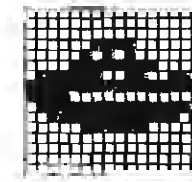
```

505	= 1179	630	= 1035	830	= 6812
510	= 9701	640	= 1824	890	= 1429
590	= 4656	690	= 483	10000	= 7545
595	= 1275	800	= 2548	10010	= 6637
600	= 691	805	= 4097		
610	= 3162	810	= 2874	TOTAL=	220145
620	= 207	820	= 3262		

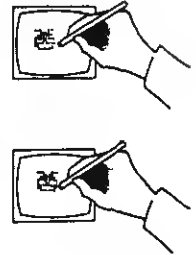
## Diseño de los sprites



CAZA



O.V.N.I.



## Tabla de variables

10	= 0	60	= 2188	300	= 3802
11	= 0	100	= 16663	305	= 6024
12	= 0	105	= 17809	310	= 715
13	= 0	110	= 7830	320	= 143
15	= 723	120	= 1865	350	= 3184
20	= 3995	130	= 1653	360	= 1181
25	= 10470	140	= 1126	370	= 143
27	= 131	200	= 3129	400	= 2235
30	= 8933	205	= 4523	405	= 2582
35	= 3590	210	= 3160	410	= 143
40	= 7700	220	= 207	450	= 9397
45	= 5489	230	= 1035	455	= 14962
50	= 6967	240	= 1599	460	= 2077
55	= 2190	290	= 593	500	= 572



# Globos

# 5



Tipo de juego:

JUEGO DE TIRO AL BLANCO

Prueba tu puntería lanzando flechas entre unos globos, intentando no estallarlos. Un buen disparo marca 10 puntos; pero ten en cuenta que sólo tienes 15 disparos.

El juego emplea una subrutina en código máquina para mover la pantalla de derecha a izquierda.

## SUGERENCIAS AL PROGRAMA

El número de lanzamientos permitidos es arbitrario; puedes cambiarlo dándole otro valor a F1 en la línea 125.

Puedes modificar este juego cambiando K-3 por K-5 en la línea 300, y el 15 por un 8 en la línea 140.

## Variables del programa

PT Puntuación.  
FL Número de flechas lanzadas.  
L Lanzamientos restantes.  
CL Control de lanzamientos.  
CF, FF Columna de flechas; fila de flechas.

		Inicialización	
5		COLOR 15,7,7 : SCREEN 1,2 : KEY OFF	
10		REM EJECUTA PROGRAMA	
11		REM SOPORTE EN CODIGO	
12		REM MAQUINA	
13		REM VER APENDICES	
15		FOR I=1072 TO 1079:READ Q:VPOKE I,Q:VPOKE I+48,Q:VPOKE 80+I,Q:NEXT:FOR I=1088 TO 1119:READ Q:VPOKE I,Q:NEXT:FOR I=1128 TO 1135:READ Q:VPOKE I,Q:NEXT:FOR I=1280 TO 1311:READ Q:VPOKE I,Q:NEXT	
16		FOR I=1216 TO 1231:READ Q:VPOKE I,Q:NEXT	
17		VPOKE 8208,135: VPOKE 8209,231: VPOKE 8210,167: VPOKE 8211,23: VPOKE 8212,231	
20		FOR I=1 TO 32: READ Q: A\$=A\$+CHR\$(Q) : NEXT : SPRITE\$(0) = A\$ : A\$ = ""	
25		FOR I =1 TO 32: READ Q: A\$=A\$+CHR\$(Q) :NEXT: SPRITE\$(1)=A\$ : A\$=""	

30		FOR I=1 TO 32: READ Q: A\$=A\$+CHR\$(Q) :NEXT: SPRITE\$(2)=A\$	
35		DEFUSR0 = 60000!: POKE 59996!,10 : POKE 59999!,8	
40		DEFUSR2=60350!: FOR I=60350! TO 60377! : READ Q : POKE I,Q : NEXT : POKE 60346!,33: POKE 60347!,24 : POKE 60348!,255 : POKE 60349!,24	
55		FOR I=6441 TO 6454:VPOKE I,134:NEXT:FOR I=6569 TO 6582:VPOKE I,134:NEXT:VPOKE 6473,134:VPOKE 6486,134:VPOKE 6537,134:VPOKE 6550,134:FOR I=1 TO 11:PRINT:NEXT	
56		PRINT TAB(7) CHR\$(134) " _ _ _ _ _ GL OBOS _ _ _ " CHR\$(134)	
60		IF INKEY\$="" THEN D=RND(1) : GOTO 60	
		Preparación pantalla	
100		CLS	
105		FOR I=6816 TO 6820 : VPOKE I,23 : NEXT : FOR I =6842 TO 6847 : VPOKE I,23 : NEXT : FOR I = 6542 TO 6790 STEP 31 : VPOKE I,162 : NEXT	
110		FOR I=6544 TO 6808 STEP 33:VPOKE I,163:NEXT:VPOKE 6821,160 :VPOKE 6841,161	
115		FOR I=6496 TO 6527 : VPOKE I,INT(RND(1)*4+136): NEXT	
120		VPOKE 6592,141: VPOKE 6590,141 :VPOKE 6600,141: VPOKE 6615,141: VPOKE 6563,141: VPOKE 6587,141	
125		FL=15: L=FL	
130		PUT SPRITE 0,(117,140) : PUT SPRITE 2,(118,139)	
135		GOSUB 1000	
140		INTERVAL ON: ON INTERVAL= 15	
140		GOSUB 550	
150		STRIG(0) ON :ON STRIG GOSUB	

```

160 800
    FOR I=6240 TO 6303 : IF RND(1)
165 <.1 THEN VPOKE I,134
    NEXT

```

Control

```

200 D=USR(D) : IF CL = 0 THEN VPO
    KE 6921, VPEEK(6913)
205 K=VPEEK(6913): IF K<= 69 THEN
    POKE 59996!,8 ELSE IF K>=165
    THEN POKE 59996!,2 ELSE POKE
    59996!,10
210 IF RND(1)<.03+PT/2000 THEN GOS
    UB 500
220 IF CL=1 THEN GOSUB 300
290 GOTO 200

```

Movimiento de flecha

```

300 K=VPEEK(6920) : VPOKE 6920,K-
    3: IF K>55 THEN RETURN

```

Comprobación de choque

```

310 IF K < 7 THEN PLAY "s1m200019
    n55n53": VPOKE 6570+FL-L,152:
    PT=PT+10: L=L-1: GOSUB 1000:
    CL=0: VPOKE 6920,139: VPOKE
    6914,0 :RETURN
320 FF = INT(K/8) : J=6144+FF*32+
    CF:K1 = VPEEK(J):IF K1 <> 134
    AND K1 <>140 THEN RETURN
330 INTERVAL OFF: VPOKE J,153: CL
    = 0: PLAY "110s10m90n20": VPO
    KE 6920,139: VPOKE 6921,VPEEK
    (6913)+1: VPOKE 6914,0: L=L-1
    : GOSUB 1000: VPOKE J,32: INT
    ERVAL ON: RETURN

```

Añadir globos

```

500 IF RND(1) <.5 THEN K1=134 ELS
    E K1=140
505 VPOKE 6207+INT(RND(1)*6)*32,
    K1
520 RETURN

```

Movimiento de pantalla

```

550 D=USR2(D) : RETURN

```

Disparar flecha

```

800 IF CL = 1 THEN RETURN
802 IF L=0 THEN 900
805 CL=1 : VPOKE 6914,4 : VPOKE
    6920,130
810 CF = INT(VPEEK(6921)/8)+1
820 RETURN

```

Fin de juego

```

900 ON INTERVAL= 7 GOSUB 550: INT
    ERVAL ON: POKE 60348!,223: PO
    KE 60349!,26
905 PUT SPRITE 0,(100,200): PUT S
    PRITE 2,(100,200): CLS: FOR I
    =1 TO 30: VPOKE 6177+ INT(RND
    (1)*700),134: PLAY "164s8m500
    n50": NEXT
910 PRINT "____TU_PUNTUACION_ES:
    ";PT
920 IF INKEY$="" THEN 930 ELSE 920
930 IF INKEY$="" THEN 930
990 SCREEN 1: END

```

# Puntuación

```
1000 FOR I = 1 TO 22: PRINT: NEXT:
PRINT "PUNTOS:";PT ; TAB(13)
; "NO. FLECHAS:";L ; CHR$(11);
1010 RETURN
```

# Datos de sprites pequeños

```
10000 DATA 120,252,252,252,120,48,
26,5
10002 DATA 0,0,0,49,123,255,255,255
10004 DATA 0,0,131,207,255,255,255,
255
10006 DATA 48,121,251,255,255,255,
255,255
10008 DATA 0,0,0,0,129,201,255,255
10010 DATA 81,213,85,54,28,8,8,8
10012 DATA 1,2,4,248,0,0,0,0
10014 DATA 128,64,32,31,0,0,0,0
10016 DATA 1,2,4,8,16,32,64,128
10018 DATA 128,64,32,16,8,4,2,1
10020 DATA 16,16,16,16,16,16,56,16
10022 DATA 146,4,32,9,128,16,69,8
```

# Control de sprites

```
10100 DATA 1,6,24,48,96,64,192,128,
128,64,32,16,8,4,2,1,128,96,
24,12,6,2,3,1,1,2,4,8,16,32,
64,128
10110 DATA 1,6,24,48,96,192,192,255
,0,0,0,0,0,0,0,0,128,96,24,12
,6,3,3,255,0,0,0,0,0,0,0,0
10120 DATA 1,3,5,1,1,1,1,1,1,1,1,
3,5,3,5,0,128,64,0,0,0,0,0,0,
0,0,0,128,64,128,64
```

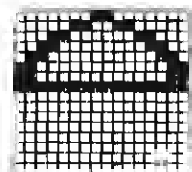
# Datos de código máquina

```
10200 DATA 42,186,235,43,35,205,27,
235,43,205,44,235,35,237,91,
188,235,123,189,194,194,235,
122,188,194,194,235,201
```

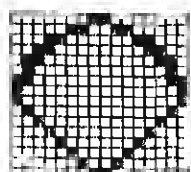
# Tabla de claves

5	= 1244	140	= 2885	910	= 2073
10	= 0	150	= 1496	920	= 1526
11	= 0	160	= 2969	930	= 1111
12	= 0	165	= 131	990	= 407
13	= 0	200	= 2497	1000	= 4954
15	= 11784	205	= 6237	1010	= 143
16	= 2155	210	= 2696	10000	= 1645
17	= 2786	220	= 982	10002	= 1448
20	= 3995	290	= 593	10004	= 1635
25	= 3996	300	= 2661	10006	= 1850
30	= 3484	310	= 9093	10008	= 1377
35	= 2481	320	= 6554	10010	= 1314
40	= 7478	330	= 10495	10012	= 1028
55	= 9435	500	= 2743	10014	= 1225
56	= 2742	505	= 2334	10016	= 1205
60	= 2297	520	= 143	10018	= 1229
100	= 159	550	= 917	10020	= 1408
105	= 6581	800	= 912	10022	= 1358
110	= 3858	802	= 860	10100	= 6404
115	= 3409	805	= 1392	10110	= 5544
120	= 4706	810	= 1976	10120	= 4771
125	= 955	820	= 143	10200	= 8518
130	= 1935	900	= 4665		
135	= 397	905	= 7875	TOTAL=	201299

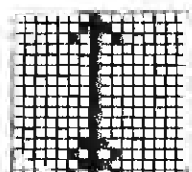
Diseño de los sprites



ARCO



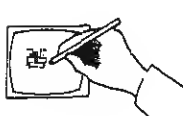
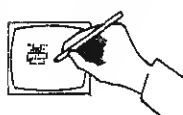
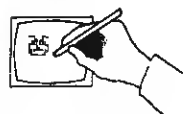
ARCO TENSO



FLECHA



GLOBO  
(CHARACTER)



# Dragster

6



Tipo de juego:

JUEGO DE CARRERAS

Conduce todo lo rápido que puedas, pero cuidate de no estrellarte con los limites de la pista o pegar un patinazo sobre el aceite.

## SUGERENCIAS AL PROGRAMA

La velocidad está controlada por el valor 9 de la línea 97. Si deseas mayor facilidad, puedes ensanchar la pista en la subrutina "Línea nueva".

## Variables del programa

D1	Distancia.
LN\$(I)	Línea a imprimir en pantalla.
I(I), D(I)	Coordenadas de la derecha e izquierda de la pista.
CA	Control de la existencia de una mancha de aceite en la pista.
C	Columna de la parte izquierda de la pista.

## Inicialización

```

10 REM EJECUTA EL PROGRAMA
11 REM SOPORTE EN CODIGO
12 REM MAQUINA
13 REM VER APENDICES ,
15 SCREEN 1,2:KEY OFF : COLOR 15
    ,4,4
20 FOR I=1 TO 32:READ Q:A$ = A$+
CHR$(Q):NEXT:SPRITE$(0)=A$:PU
T SPRITE 0,(123,0),1
22 A$="":FOR I=1 TO 8:READ Q:A$=
A$+CHR$(Q):NEXT:SPRITE$(1)=A$
25 DEFUSR=60000!:DEFUSR1=60118!:
POKE 59996!,10:POKE 59999!,3:
POKE 59997!,0:POKE 59998!,0
27 FOR I=60300! TO 60331!:READ Q
:POKE I,Q:NEXT:DEFUSR2=60300!
:POKE 60296!,223:POKE 60297!,
26:POKE 60298!,0:POKE 60299!,
24
30 PRINT "_____***_____*":P
RINT "___DA_A___***_____*":PR
INT "_____***_____*":PRI
NT "CUALQUIER_***_D_***":PRIN
T "___TECLA___***_R_***":PRINT
"_____***_A_***"
35 PRINT "___PARA___***_G_***":P

```

```

PRINT "EMPEZAR_***S_***":PR
INT "*****T_***":PRIN
T "*****E_***":PRIN
T "*****R_***":PRINT
"*****"
PRINT "*****_***":
PRINT "*****_***":
PRINT "*****_***":
PRINT "*****_***":
PRINT "*****_***":
PRINT "*****_***":
PRINT "*****_***":
FOR I =1 TO 5:PRINT "*****
_***_***": NEXT: PRINT "*****
*****";
IF INKEY$<>" THEN 60
D=RND(1):D=USR1(D):GOTO 45

```

### Preparación pantalla

```

60 PRINT :PUT SPRITE 0,(123,176)
   ,1:FOR I=1 TO 23:PRINT"_____
   ____**____**": NEXT
65 PRINT CHR$(11);:FOR I=1 TO 6:
PRINT: NEXT: PRINT "____+~+~"
   : FOR I=1 TO 7: PRINT "____!
   ~!": NEXT:PRINT"____+~+~": FO
   R I=1 TO 7 : PRINT "____!":
   NEXT
70 VPOKE 6440,51:VPOKE 8208,149
75 FOR I=1 TO 800: NEXT :VPOKE
   6504,50:VPOKE 8208,164:VPOKE
   6440,32:FOR I=1 TO 800: NEXT:
   VPOKE 6504,32:VPOKE 6568,49:
   VPOKE 8208,36
80 IF INKEY$="" THEN 80 ELSE POK
   E 59999!,2:PLAY"12s8m40000n2"
85 D=USR1(D): IF VPEEK (6912)>25
   THEN 85
90 PRINT CHR$(11);: FOR I=1 TO 4
   :LN$(I)="_____**____**
   _____": PRINT LN$(I);:
   PRINT "_____
   _____":I(I)=110:D(I)=136:NEXT

```

```

95 POKE 59999!,7: C=11:ON SPRITE
GOSUB 900: SPRITE ON
97 ON INTERVAL=9 GOSUB 300: INTE
RVAL ON
98 FOR I=152 TO 175 STEP 2: VPOKE
E I,170: VPOKE I+1,85: NEXT:
VPOKE 8194,228

```

No hacer nada

```

100 GOTO 100

```

Nueva linea

```

200 FOR I=4 TO 2 STEP-1: LN$(I)=L
N$(I-1): D(I)=D(I-1): I(I)=I(
I-1):NEXT
210 I=INT(RND(1)*3-1): C=C+I: IF
C<2 THEN C=2 ELSE IF C>12 THE
N C=12
215 LN$(1)="
*****": MID$(LN$(1),C)="*
**.....***": I(1)=8*C+28: D
(1)=I(1)+48
220 IF RND(1)<.05+DI/1000 AND CA=
0 THEN PUT SPRITE 1,(I(1)+INT
(RND(1)*35+5),0),1:CA=1
225 IF VPEEK(6916)>190 THEN PUT S
PRITE 1,(200,200):CA=0
250 RETURN

```

Control

```

300 D=USR2(D)
305 GOSUB 200:PRINT CHR$(11);LN$(
1):PLAY"110m59000s10n4"
310 IF CA=1 THEN VPOKE 6916,VPEEK
(6916)+8
320 DI=DI+1
330 GOSUB 800

```

```

340 D=USR(D)
350 RETURN

```

Comprobación de choque

```

800 IF VPEEK(6913)<I(4) OR VPEEK(
6913)>D(4) THEN 900
810 RETURN

```

Fin de juego

```

900 INTERVAL OFF:PLAY"11m60000s8n
20":FOR I=6144 TO 6303:VPOKE
I,32:NEXT:PRINT CHR$(11);"E_L_S
_L_T_R_E_L_L_L_A_D_D_O_!_!_!":PRINT
:PRINT"DISTANCIA: ";DI;"KM"
905 FOR I=1 TO 2500: NEXT: SCREEN
0
910 IF INKEY$="" THEN END ELSE
910

```

Datos de sprites

```

10000 DATA 57,63,59,3,3,3,3,3,3,
247,247,255,243,240,240,156,
252,220,192,192,192,192,192,
192,192,239,239,255,207,15,15
10010 DATA 112,236,88,248,204,0,0,0

```

Datos de movimiento

```

10100 DATA 42,136,235,35,43,205,27,
235,17,32,0,25,205,44,235,237
,82,237,91,138,235,123,189,
194,144,235,122,188,194,144,
235,201

```

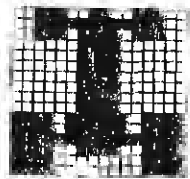


## Tabla de claves

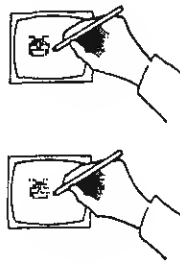
10	= 0	70	= 955	310	= 1938
11	= 0	75	= 5846	320	= 798
12	= 0	80	= 3306	330	= 158
13	= 0	85	= 2173	340	= 691
15	= 1220	90	= 11071	350	= 143
20	= 4519	95	= 2355	800	= 2924
22	= 4010	97	= 2767	810	= 143
25	= 5685	98	= 3578	900	= 10672
27	= 7288	100	= 489	905	= 1417
30	= 12701	200	= 5312	910	= 1449
35	= 13501	210	= 5383	10000	= 9121
40	= 16889	215	= 7923	10010	= 1462
42	= 3504	220	= 6661	10100	= 10025
45	= 1243	225	= 3063		
50	= 2186	250	= 143		
60	= 3483	300	= 716		
65	= 8199	305	= 2930		

TOTAL= 190040

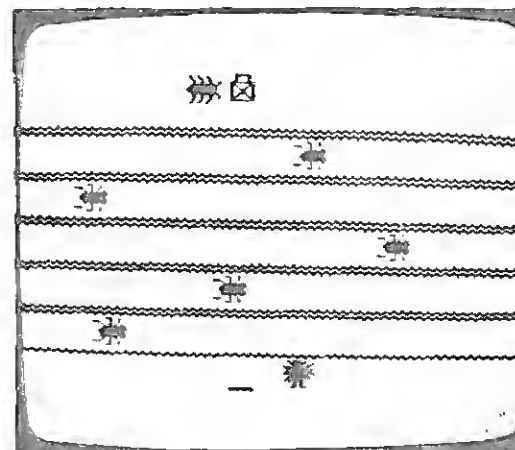
## Diseño de los sprites



COCHE



# Las cucarachas 7 cósmicas



tipo de juego:

JUEGO DE EVASION

Rigoberto, "el Hambriento", peludo hombre de las montañas, ha llegado al dominio de las cucarachas cósmicas en busca de alimento. Conduce a Rigoberto a través de los túneles para que llegue hasta la cucaracha reina, y le quite parte de la comida que guarda. Cuando lo logre, todavía ha de volver al campamento.

Haz todos los viajes que puedas hasta que las cucarachas cósmicas te atrapen.

## SUGERENCIAS AL PROGRAMA

En la línea 210 se fija la frecuencia con que se acercan las cucarachas cósmicas a Rigoberto; haz los cambios que creas convenientes para aumentar o disminuir la dificultad. Si la rapidez de las cucarachas no te emociona, cámbiala en la línea 250.

### Variables del programa

CC Cucaracha más cercana a Rigoberto.  
DC(I) Dirección de las cucarachas.  
NV Número de viajes completos que ha hecho Rigoberto.  
CO ¿Ha cogido comida Rigoberto?

		Inicialización	
5	CLEAR 700 : COLOR 1,14,14 : S		
	CREEN 2,2		
10	REM EJECUTA PROGRAMA		
11	REM SOPORTE EN CODIGO		
12	REM MAQUINA		
13	REM VER APENDICES		
15	FOR I=1 TO 32: READ Q: A\$ =A\$		
	+ CHR\$(Q): NEXT : SPRITE\$(0)=		
	A\$: A\$=""		
20	FOR I=1 TO 32: READ Q: A\$= A\$		
	+ CHR\$(Q): NEXT: FOR I=1 TO 5		
	: SPRITE\$(I)=A\$: NEXT:A\$=""		
25	FOR I=1 TO 32: READ Q: A\$=A\$+		
	CHR\$(Q): NEXT: SPRITE\$(6)=A\$		
30	DEFUSR=60000!: DEFUSR1=60118!		
	: POKE 59996!,15		
35	PUT SPRITE 0,(120,173),1: PUT		
	SPRITE 2,(255,72),12: PUT SPR		
	ITE 3,(255,98),12: PUT SPRITE		
	4,(255,124),12: PUT SPRITE 5,		
	(255,150),12 : PUT SPRITE 6,(		
	255,46),1		
40	FOR I=1 TO 6: IF RND(1)<.5 TH		
	EN DC(I)=1 ELSE DC(I)=3		
45	NEXT :CC=5		
50	B\$="e1f2e1":A\$="": FOR I = 1		

52	TO 32 : A\$=A\$+B\$ : NEXT		
	DRAW "bm0,41xa\$;xa\$;bm0,44xa\$		
	;xa\$;"		
54	DRAW "bm0,67xa\$;xa\$;bm0,70xa\$		
	;xa\$;"		
56	DRAW "bm0,93xa\$;xa\$;"		
58	DRAW "bm0,96xa\$;xa\$;"		
60	DRAW "bm0,119xa\$;xa\$;"		
62	DRAW "bm0,122xa\$;xa\$;"		
64	DRAW "bm0,145xa\$;xa\$;"		
66	DRAW "bm0,148xa\$;xa\$;"		
68	DRAW "bm0,171xa\$;xa\$;"		
70	GOSUB 78		
75	DRAW "bm115,192r10u2110"		
77	GOTO 90		
78	DRAW "bm115,14r10d10110u10f10		
	b110e10b18u4r6d4":RETURN		
79	DRAW"bm115,178r10d10110u10f10		
	b110e10b18u4r6d4":RETURN		
90	SPRITE ON : ON SPRITE GOSUB		
	800		
		Control	
100	POKE 59999!,10: D=USR(D): GOS		
	UB 300		
110	GOSUB 200		
190	GOTO 100		
		Ajuste dirección de cucarachas	
200	GOSUB 250		
205	I=INT(RND(1)*6+1): IF DC(I)=1		
	THEN DC(I)=3 ELSE DC(I)=1		
210	IF RND(1)<.9-NV/100 THEN 220		
	ELSE IF VPEEK(6913+4*CC)<VPEE		
	K(6913) THEN DC(CC)=3 ELSE DC		
	(CC)=1		
220	RETURN		
		Movimiento de cucarachas	
250	POKE 59999!,9: FOR I =1 TO 6:		

```

260      POKE 59998!,DC(I):POKE 59997!
      ,I: D = USR1(D): NEXT
      RETURN

```

Comprobación de movimiento

```

300      K=VPEEK(6912): CC=INT((K-20)/
26)
305      IF K<72 THEN CC=5
320      IF K>180 THEN K1=11 ELSE IF K
<10 THEN K1=14 :PSET K1 =15
325      POKE 59996!,K1
330      IF K>25 AND K<165 THEN RETURN
335      IF VPEEK(6913)<115 OR VPEEK(
6913)>125 THEN RETURN
340      IF K<=25 THEN IF CO=1 THEN RE
TURN ELSE PLAY "124m160s8n60n
60":CO=1 : DRAW "c14" : GOSUB
78 :GOSUB 79 : RETURN
345      IF CO=1 THEN PLAY"s8m200140n2
0" : NV= NV+1 : CO=0: DRAW "c
1" : GOSUB 79: GOSUB 78 : RE
TURN
350      RETURN

```

Fin de juego

```

800      FOR I=1 TO 7 : PLAY "150m1300
s10n64r10n74": NEXT
810      SCREEN 1: PRINT "____LAS_CUCAR
ACHAS_COSMICAS_____ATACA
N_DE_NUEVO!"
820      PRINT:PRINT:PRINT:IF NV>1 THE
N PRINT "PERO__PEPE_EL_ALPINI
STA_": PRINT"HA_HECHO_";NV;"V
IAJES_ALPOR_____COMIDA"
821      IF NV=1 THEN PRINT "____P
OBRE_PEPE_": PRINT:PRINT "____
SOLO_HIZO_UN_VIAJE" ELSE PR
INT: PRINT:PRINT "____PEPE_N
O_TIENE_COMIDA!"
850      IF INKEY$="" THEN END ELSE
850

```

Datos de sprites

```

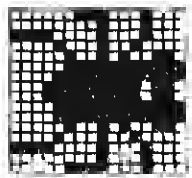
10000      DATA 18,10,5,82,47,21,37,87,
14,23,39,75,18,2,2,14,144,161
,74,148,104,178,164,234,113,
232,228,208,64,64,64,112
10010      DATA 3,0,240,8,4,7,15,31,31,
15,15,4,8,240,0,3,136,136,136
,144,166,255,255,246,246,255,
255,166,144,136,136,136
10020      DATA 68,34,17,17,17,63,127,
255,255,127,63,17,17,17,34,68
,64,32,16,17,17,250,252,252,
252,252,250,17,17,16,32,64

```

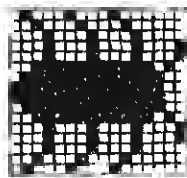
## Tabla de claves

5	= 1169	64	= 1442	305	= 1186
10	= 0	66	= 1438	320	= 3455
11	= 0	68	= 1441	325	= 786
12	= 0	70	= 238	330	= 1621
13	= 0	75	= 1522	335	= 2727
15	= 3995	77	= 483	340	= 5993
20	= 5120	78	= 3573	345	= 5170
25	= 3484	79	= 3641	350	= 143
30	= 2747	90	= 926	800	= 2820
35	= 8795	100	= 1837	810	= 4617
40	= 3711	110	= 362	820	= 7792
45	= 588	190	= 489	821	= 8567
50	= 3498	200	= 410	850	= 1397
52	= 2587	205	= 4600	10000	= 8605
54	= 2595	210	= 7519	10010	= 8600
56	= 1385	220	= 143	10020	= 8785
58	= 1398	250	= 4786		
60	= 1435	260	= 143	TOTAL=	153842
62	= 1438	300	= 2650		

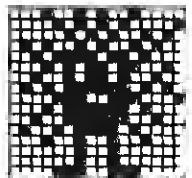
## Diseño de sprites



CUCARACHA



CUCARACHA REINA

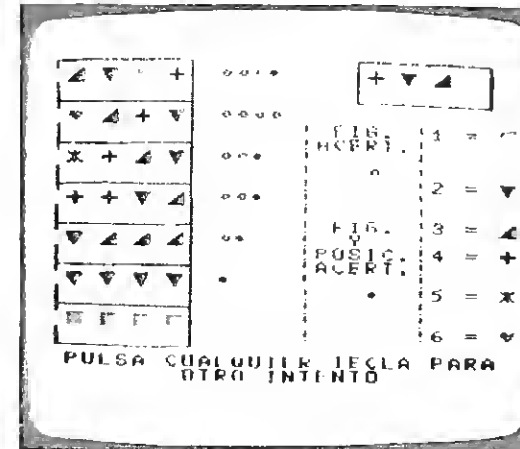


RIGOBERTO,  
«EL HAMBRIENTO»



# Test de inteligencia

# 8



Tipo de juego:

JUEGO MEMORISTICO

El código secreto está formado por la combinación de cuatro figuras, elegidas de seis tipos diferentes. Adivina el código en menos de ocho intentos.

Un círculo significa que tienes una figura correcta, pero en una posición equivocada: un círculo coloreado te indica que tienes una figura en la posición correcta.

Las instrucciones van incluidas en el programa.

## SUGERENCIAS AL PROGRAMA

Si encuentras algo difícil el juego puedes tener más oportunidades, cambiando la rutina "Juego perdido" por otra que dé al jugador más facilidades o se rinda.

Por otra parte, puedes añadir otra figura, con el objetivo de aumentar el nivel de dificultad.

## Variables del programa

- CL(I) Código de caracteres de la clave secreta.
- CJ(I) Código de caracteres de la clave propuesta en una baza por el jugador.
- N1 Nivel (0-6).
- C Columna (0-3).
- M\$ Mensaje.
- FC Número correcto de figuras.
- FA Número de figuras acertadas en forma y posición.

		Inicialización	
5	SCREEN 1 : KEY OFF :COLOR 1,		
	15,15 : PRINT".....TEST_DE_IN		
	TELIGENCIA"		
10	VPOKE 8209,79: VPOKE 8210,207		
	: VPOKE 8212,111 : VPOKE 8214		
	,159 : FOR I= 136 TO 184 STEP		
	8: FOR J=0 TO 7: READ Q: VPOK		
	E 8*I+J,Q : NEXT : NEXT		
15	FOR I=1 TO 5 : PRINT : NEXT :		
	PRINT ".....UTILIZA LAS TECLAS_D		
	E_LA.....1' _A_LA.....6' _PARA_PONE		
	R_EN.....LA_PANTALLA LAS_6_FIGU		
	RAS.....POSIBLES.": PRINT "_LA		
	_TECLA _DEL' _FUEDE_USARSE _PAR		
	A_CAMBIAR_LA_ULTIMA_FIGU		
20	PRINT: PRINT"....."CHR\$(136)".....		
	....."CHR\$(144)"....."CHR\$(152)".....		
	"CHR\$(160)"....."CHR\$(168)"....."		
	CHR\$(176)		
30	PRINT : PRINT : PRINT "PULSA		
	UNA_TECLA_PARA_EMPEZAR"		
35	IF INKEY\$="" THEN D=RND(1): G		

OTO 35

Preparación pantalla

100	CLS : FOR I=1 TO 6 :PRINT : N	
	EXT : PRINT TAB(16) "!_FIG._!"	
	"TAB(24) "1_=_ " CHR\$(136);	
105	PRINT TAB(16) "!ACERT.!" :PRINT	
	TAB(16) "!.....!" : PRINT TAB(	
	16) "!....."CHR\$(248) ".....!" :PRINT	
	TAB(16) "!.....!" TAB(24) "2_=_	
	_ "CHR\$(144) TAB(16) "!.....!"	
106	PRINT TAB(16) "!.....!" :PRINT	
	TAB(16) "!_FIG._!"TAB(24) "3_=_	
	_ "CHR\$(152);	
110	PRINT TAB(16) "!_Y.....!" : PRI	
	NT TAB(16) "!POSIC.!"TAB(24) "4	
	_=_ "CHR\$(160);:PRINT TAB(16) "	
	!ACERT.!" : PRINT TAB(16) "!.....	
	.....!" : PRINT TAB(16) "!....."CH	
	R\$(249) ".....!" TAB(24) "5_=_ "CHR	
	\$(168);	
115	PRINT TAB(16) "!.....!" : PRIN	
	T TAB(16) "!.....!" : PRINT T	
	AB(16) "!.....!"TAB(24) "6_=_	
	"CHR\$(176)CHR\$(11);	
140	VPOKE 6209,84:VPOKE 6241,69:V	
	POKE 6273,83:VPOKE 6305,84:VP	
	OKE 6369,68:VPOKE 6401,69:VPO	
	KE 6465,73:VPOKE 6497,70:VPOK	
	E 6529,84	
141	VPOKE 6561,69:VPOKE 6593,76:V	
	POKE 6625,73:VPOKE 6657,71:VP	
	OKE 6689,69:VPOKE 6721,70:VPO	
	KE 6753,67:VPOKE 6785,73:VPOK	
	E 6817,65	
145	FOR I=6179TO 6755 STEP 96: F	
	OR J=I+97 TO I+103: VPOKE J,	
	23: NEXT: VPOKE I,20: VPOKE I	
	+8,19: VPOKE I=32,22: VPOKE I	
	+64,22: VPOKE I+40,22: VPOKE	
	I+72,22: NEXT	
150	VPOKE 6179,24: VPOKE 6187,25:	
	VPOKE 6851,26: VPOKE 6859,27:	

```

FOR I=6180 TO 6186: VPOKE I,
23: NEXT
155 FOR I= 6197 TO 6261 STEP32: F
OR J= 0 TO 6 : VPOKE I+J,184:
NEXT: NEXT: FOR I=1 TO 4
160 K=INT(RND(1)*6): IF T(K)<>1
THEN CL(I)=K*8+136: T(K)=1 EL
SE 160
165 NEXT: M$="LISTO_CUANDO QUIERA
S": GOSUB 1000

```

Editor

```

200 X$=INKEY$: IF X$<>CHR$(127)A
ND(X$<"1" OR X$>"6") THEN 200
205 IF X$=CHR$(127) THEN IF C=0 T
HEN 200 ELSE C=C-1:VPOKE 6788
+2*C-96*L,32:GOTO 200
210 CJ(C+1)=VAL(X$)*8+128: VPOKE
6788+2*C-96*L,CJ(C+1):C=C+1
215 IF C=4 THEN M$="COMPROBANDO_T
U_RESPUESTA": GOSUB 1000: FOR
I=1 TO 1500: NEXT: GOTO 300
290 GOTO 200

```

Comprobación jugada

```

300 FC=0:FA=0:FOR I=1 TO 4
305 IF CJ(I)=CL(I) THENFA=FA+1
310 NEXT: FOR I=1 TO 4
315 IF CL(I)=CJ(1) OR CL(I)=CJ(2
) OR CL(I)=CJ(3) OR CL(I)=CJ(
4) THEN FC=FC+1
320 NEXT: FC=FC-FA

```

Listar resultados

```

400 IF FC=0 THEN 415
410 FOR I=1 TO FC: VPOKE 6796-96*
L+I,248: NEXT
415 IF FA=0 THEN 425

```

```

420 FOR I=1 TO FA: VPOKE 6796+FC-
96*L+I,249: NEXT
425 C=0:L=L+1:M$="
": GOSUB 1000: IF FA
=4 THEN 900
430 IF L=7 THEN 500
435 GOTO 200

```

Juego perdido

```

500 FOR I=6197 TO 6261 STEP 32: F
OR J=0 TO 6: VPOKE I+J,32: NE
XT: NEXT: FOR J=6198 TO 6204:
VPOKE J,23: NEXT: VPOKE6197,
24: VPOKE 6205,25: VPOKE 6229
,22: VPOKE 6261,22:VPOKE 6237
,22: VPOKE 6269,22
505 VPOKE 6293,26: VPOKE 6301,27:
FOR I=6294 TO 6300:VPOKE I,23
:NEXT
510 FOR I=1 TO 4: PLAY"n20s8m160n
30": VPOKE 6228+2*I,CL(I): NE
XT
515 FOR I=1 TO 2500: NEXT: M$="_P
ULSA_CUALQUIER_TECLA_PARA_
OTRO_INTENTO": GOSUB
1000
520 IF INKEY$="" THEN 520 ELSE RU
N

```

Juego ganado

```

900 M$="-----SUPER!!!!": GO
SUB 1000: FLAY"s14m2000110n30
n32n34n36n38n40n42n44m1000n46
n48n49n50n51m700n52n53n54n55m
500m500n56n58n60"
910 FOR I=1 TO 3000: NEXT: M$="LD
_HAS_LOGRADO_EN"+STR$(L)+"_IN
TENTOS": GOSUB 1000: BEEP
920 FOR I=1 TO 3000: NEXT:M$="_PU
LSA_UNA_TECLA_PARA_JUGAR_
OTRA_VEZ": GOSUB 1000 :

```



930	BEEP IF INKEY\$="" THEN 930 ELSE RU N
	Listas mensaje
1000	FOR I=1 TO 22: PRINT: NEXT: PRI
1010	NT M\$; CHR\$(11); RETURN
	Datos de sprites pequeños
10000	DATA 255,255,255,255,255,255, 255,255
10002	DATA 255,255,126,126,60,60,24 ,24
10004	DATA 1,3,7,15,31,63,127,255
10006	DATA 24,24,24,255,255,24,24, 24
10008	DATA 146,84,56,16,56,84,146,0
10010	DATA 108,254,254,254,124,56, 16,0
10012	DATA 255,255,255,255,255,255, 255,255

920	= 6473	10000	= 1888	10008	= 1469
930	= 1482	10002	= 1654	10010	= 1645
1000	= 2190	10004	= 1326	10012	= 1888
1010	= 143	10006	= 1515	TOTAL	= 236456

## Tabla de claves

5	= 3659	150	= 4124	400	= 941
10	= 6786	155	= 4353	410	= 2911
15	= 21816	160	= 5511	415	= 951
20	= 6544	165	= 2839	420	= 3325
30	= 3010	200	= 4410	425	= 4623
35	= 2286	205	= 6073	430	= 977
100	= 3978	210	= 5600	435	= 593
105	= 11686	215	= 5937	500	= 14774
106	= 2440	290	= 593	505	= 556
110	= 12949	300	= 1653	510	= 3738
115	= 5610	305	= 2072	515	= 6696
140	= 6491	310	= 910	520	= 1320
141	= 6566	315	= 5970	900	= 12358
145	= 9819	320	= 1115	910	= 6220

# El saltarín del río

# 9



Tipo de juego:

JUEGO DE SIMULACION

La vida es dura para la pequeña rana que quiere cruzar el río. Cruza el río todas las veces que puedas; la puntuación depende del número total de movimientos que emplees para cruzarlo cada vez.

Se emplea una subrutina en código máquina para mover la pantalla.

## SUGERENCIAS AL PROGRAMA

La corriente del río se controla por el valor 20 en la línea 150; para acelerarla, cambia el 20 por 10. La densidad de los nenúfares está controlada por los valores 0.65 y 0.60 en las líneas 125 y 135, respectivamente.

¿Por qué no tener otros objetos flotantes diferentes en una de las mitades del río? Puedes poner hasta un barquito (como un sprite).

## Variables del programa

D	Direcciones de la rana (0 ó 2).
PR	Posición de la rana.
NM	Número de movimiento.
PT	Puntuación.
FJ, CJ	Fila y columna del jugador.
RS	Objeto sobre el que está la rana
LT	Controlador de llegada al tablón.

		Inicialización	
	5	REM EJECUTA EL CODIGO	
	6	REM MAQUINA SOPORTE	
	7	REM DEL PROGRAMA	
	8	REM VER APENDICES	
	10	COLOR 4,7,1: SCREEN 1,2: KEY OFF	
	15	FOR I=60350! TO 60401!: READ Q: POKE I,Q: NEXT	
	20	FOR I=1072 TO 1079: READ Q: VPOKE I,Q: NEXT: FOR I=1104 TO 1135: READ Q: VPOKE I,Q: NEXT: FOR I=1152 TO 1175: READ Q: VPOKE I,Q: NEXT: FOR I=1216 TO 1223: READ Q: VPOKE I,Q: NEXT	
	25	FOR I=1 TO 11: PRINT: NEXT: PRINT "PULSA UNA TECLA"	
	35	PLAY"124m160s8n67n67":PRINT CHR\$(11)TAB(8)"SALTARIN-DEL-RIO": FOR I=1 TO 300: NEXT	
	40	VPOKE 6208+INT(RND(1)*60),134	

		:IF INKEY\$="" THEN FOR I=6148 TO 6170: VPOKE I,32: NEXT: FOR I=1 TO 300: NEXT: GOTO 35	
50		DEFUSR=60350! : SA=6240 : FA=6783: POKE 60346!,SA MOD 256: POKE 60347!,INT(SA/256): POKE 60348!,FA MOD 256:POKE 60349!,INT(FA/256)	
		Preparación pantalla	
	100	CLS: FOR I=6155 TO 6207: VPOKE I,141: NEXT: VPOKE 6144,141: VPOKE 6145,141	
	105	FOR I=6208 TO 6239 : VPOKE I,140: NEXT: FOR I=6784 TO 6815 : VPOKE I,139: NEXT	
	110	FOR I=6816 TO 6911 : VPOKE I,141: NEXT	
	115	FOR I=6496 TO 6525: J =INT(RND(1)*3+144): VPOKE I,J: NEXT	
	120	GOSUB 1000	
	125	FOR I=6240 TO 6464 STEP 32: F OR J=0 TO 31: IF RND(1)<.65 THEN VPOKE I+J,138	
	130	NEXT: NEXT	
	135	FOR I=6528 TO 6752 STEP 32: F OR J=0 TO 31: IF RND(1)<.6 THEN VPOKE I+J,138	
	140	NEXT: NEXT	
	145	RS=139: PR=6799: VPOKEPR,134: CJ=15	
	150	ON INTERVAL=20 GOSUB 800: INTERVAL ON	
	155	FOR I=1 TO 5: VPOKE 6176+INT(RND(1)*31),152: NEXT: FOR I=1 TO 12: VPOKE 6816+INT(RND(1)*95),152: NEXT	
	160	VPOKE 8208,19: VPOKE 8209,55: VPOKE 8210,231:VPOKE 8211,179	
		Control	
	200	X\$=INKEY\$: IF X\$<>"" THEN GOS	

```

210 UB 300
290 IF RND(1)<.1 THEN GOSUB 400
    GOTO 200

```

Leer del teclado

```

300 FJ=INT((PR-6144)/32): CJ=PR-
32*FJ-6144
305 K=ASC(X$)-27: IF K>0 AND K<5
    THEN PLAY"124m160s8n67":ON K
    GOTO 310,315,320,325 ELSE RET
    URN
310 IF CJ=31 THEN RETURN ELSE INT
    ERVAL OFF: VPOKE PR,RS:PR=PR+
    1: GOTO 340
315 IF CJ=0 THEN RETURN ELSE INTE
    RVAL OFF:VPOKE PR,RS: PR=PR-1
    : GOTO 340
320 IF FJ<=2 THEN RETURN ELSE INT
    ERVAL OFF:VPOKE PR,RS: PR=PR-
    32: GOTO 340
325 IF FJ>=20 THEN RETURN ELSE IN
    Terval OFF:VPOKE PR,RS: PR=PR
    +32
340 NM=NM+1: RS=VPEEK(PR): VPOKE
    PR,134:INTERVAL ON: IF RS=32
    THEN 900 ELSE 500

```

Cambiar pantalla

```

400 IF RND(1)<.5 THEN I=6240 ELSE
    I=6528
405 VPOKE I+32*(INT(RND(1)*8)),
    138: VPOKE I+32*(INT(RND(1)*8
    )),32
450 RETURN

```

Comprobación de posición

```

500 IF D=2 THEN 520
505 IF FJ=2 THEN PT=PT+200-NM: D=

```

```

2: LT=0: GOSUB 1000: NM=0: RE
    TURN
510 IF FJ=11 AND LT=0 THEN LT=1:
    PT=PT+50: GOSUB 1000
515 RETURN
520 IF FJ=20 THEN D=0: PT=PT+200-
    NM: LT=0: GOSUB 1000: NM=0: R
    ETURN
525 GOTO 510

```

Movimiento de pantalla

```

800 D=USR(D): IF PR<=6783 AND PR>=
    6240 THEN PR=PR-1 ELSE RETURN
810 IF PR/32=INT(PR/32) THEN 900
    ELSE RETURN

```

Fin de juego

```

900 CLS: FOR I=1 TO 50: PLAY"164s
    14m50n50": VPOKE 6176+INT(RND
    (1)*700),134: NEXT
910 PRINT CHR$(11); "PUNTOS: ";PT
990 END

```

Puntuación

```

1000 PLAY"12s10m700n44": PRINT "PU
    NTOS: ";PT;CHR$(11);
1010 RETURN

```

Datos del código máquina

```

10000 DATA 6,31,42,186,235,43,35,
    205,27,235,120,254,31,202,215
    ,235,4,43,205,44,235,35,195,
    227,235,6,0,17,31,0,25,205,44
    ,235,183,237,82,237,91,188,

```

235, 123, 189, 194, 196, 235, 122,  
188, 194, 196, 235, 201

Datos de sprites pequeños

10100 DATA 0, 36, 90, 255, 255, 189, 66, 60  
10102 DATA 60, 124, 254, 255, 255, 255,  
126, 124  
10104 DATA 0, 131, 207, 255, 255, 255,  
255, 255  
10106 DATA 255, 255, 255, 255, 255, 247,  
97, 0  
10108 DATA 255, 255, 255, 255, 255, 255,  
255, 255  
10110 DATA 32, 255, 255, 255, 255, 255,  
255, 11  
10112 DATA 8, 255, 255, 255, 255, 255,  
255, 1  
10114 DATA 97, 251, 255, 255, 255, 255,  
255, 204  
10116 DATA 56, 56, 254, 254, 214, 16, 16,  
16

1000 = 3594  
1010 = 143  
10000 = 21635  
10100 = 1538  
10102 = 1836

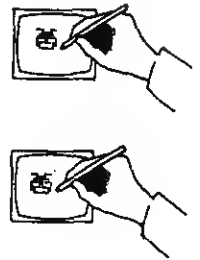
10104 = 1775  
10106 = 1712  
10108 = 1888  
10110 = 1781  
10112 = 1661

10114 = 1857  
10116 = 1587  
TOTAL = 188709

## Diseño de los sprites



RANA  
(CARACTER)



## Tabla de claves

5	= 0	125	= 4578	325	= 4277
6	= 0	130	= 320	340	= 5791
7	= 0	135	= 4642	400	= 2613
8	= 0	140	= 320	405	= 4913
10	= 1228	145	= 1071	450	= 143
15	= 2217	150	= 2742	500	= 733
20	= 9678	155	= 7006	505	= 4582
25	= 3079	160	= 2050	510	= 3407
35	= 5601	200	= 1981	515	= 143
40	= 7011	210	= 1587	520	= 4592
50	= 9403	290	= 593	525	= 643
100	= 3158	300	= 3512	800	= 4363
105	= 3798	305	= 6014	810	= 2554
110	= 1966	310	= 4716	900	= 4985
115	= 3923	315	= 4667	910	= 1639
120	= 397	320	= 4937	990	= 129

# Prueba de fútbol

# 10



Tipo de juego:

JUEGO DE TIRO AL BLANCO

¿Cuánto tiempo puedes mantener el balón en el aire?

Intenta que el balón dé en los agujeros del techo; son puntos extra. Puedes dar cabezazos o patadas; para las patadas usa la barra espaciadora.

## SUGERENCIAS AL PROGRAMA

Es fácil cambiar la velocidad del balón —mira VB en la línea 80—. Te sugiero que pongas algunos obstáculos en la pared para que cuando el balón choque con ellos rebote con un ángulo aleatorio.

Revisa las líneas 700 a 750 para poder golpear el balón con la rodilla o poder dar otros ángulos al golpearla.

### Variables del programa

DR	Dirección del balón (0-7).
PT	Puntuación.
X, Y	Coordenadas del balón.
JX	Coordenada X del jugador.
VB	Velocidad del balón.
GA	Golpe en el agujero del techo (1-3).
GJ	Control de golpe del balón sobre el jugador.

#### Inicialización

```

10 REM EJECUTA EL PROGRAMA
11 REM SOPORTE EN CODIGO MAQUINA
12 REM VER APENDICES
15 SCREEN 1,3:KEY OFF:COLOR 15,3
   ,3:PRINT "FUTBOL"
20 FOR I=1 TO 32: READ Q: A$=A$+
  CHR$(Q): NEXT: SPRITE$(0)=A$:
  A$=""
25 FOR I=1 TO 32: READ Q: A$=A$+
  CHR$(Q):NEXT: SPRITE$(1)=A$:A$
  =""
30 FOR I=1 TO 32: READ Q: A$=A$+
  CHR$(Q): NEXT: SPRITE$(2)=A$:
  A$=""
35 FOR I=1 TO 8:READ Q:A$=A$+CHR
  $(Q):NEXT:SPRITE$(3)=A$
40 DEFUSR=60000!:DEFUSR1=60118!:
  POKE 59996!,10
45 FOR I=1 TO 20:PRINT:NEXT:PRIN
  T"PULSA UNA TECLA":PO
  KE 59997!,3:POKE 59998!,3:POK

```

```

47 E 59999!,1:PUT SPRITE 0,(120,
   100),1:PUT SPRITE 3,(0,20)
   FOR I=1088 TO 1103:READ Q:POK
   E I,Q:NEXT
50 D=USR1(D): IF VPEEK(6925)=255
   THEN POKE 59998!,1 ELSE IF VP
   EEK(6925)=1 THEN POKE 59998!,
   3
55 IF INKEY$="" THEN 50

```

#### Preparación pantalla

```

60 CLS:K=134:VPOKE 1072,170:VPOK
  E 1073,127:VPOKE 1074,254:VPO
  KE 1075,127:VPOKE 1076,254:VP
  OKE 1077,127:VPOKE 1078,254:V
  POKE 1079,85: FOR I=6144 TO
  6175: VPOKE I,K: NEXT
61 VPOKE 6147,32: VPOKE 6148,32:
  VPOKE 6159,32: VPOKE 6160,32:
  VPOKE 6172,32: VPOKE 6173,32
65 FOR I=6176 TO 6880 STEP 32:VP
  OKE I,K:VPOKE I+31,K:NEXT:FOR
  I=6849 TO 6879:VPOKE I,K:NEXT
70 FOR I=6893 TO 6911:VPOKE I,K:
  NEXT
75 VPOKE 8208,195:GOSUB 1000:PUT
  SPRITE 0,(120,142):PUT SPRITE
  3,(11,11)
80 DR=6:VB=7:STRIG(0) ON:ON STRI
  G GOSUB 800
85 K=6817:FOR I=1 TO 11:VPOKE K,
  29:VPOKE K+16,29:K=K-31:NEXT:
  FOR I=6465 TO 6494:VPOKE I,23
  :VPOKE I-96,23:NEXT
95 SPRITE ON :ON SPRITE GOSUB
  700

```

#### Control

```

100 POKE 59999!,6:D=USR(D)
105 IF PEEK(59997!)=0 THEN POKE
  6914,0

```

```

110 GOSUB 200
120 GOSUB 300
190 GOTO 100

```

Movimiento del balón

```

200 IF DR>3 THEN SPRITE OFF:GOTO
210:ELSE POKE 59999!,VB:POKE
59997!,3:POKE 59998!,DR:D=USR
1(D):GOTO 220
210 POKE 59997!,3:POKE 59999!,VB-
1:K1=DR-4:K2=(DR-3) MOD 4:POK
E 59998!,K1:SPRITE ON :D=USR1
(D):POKE 59998!,K2:D=USR1(D)
220 X=VPEEK(6925): Y=VPEEK(6924):
RETURN

```

Comprobación de posición

```

300 IF Y<8 OR Y>192 THEN GJ=0:PLA
Y "19≤14m1900n30":GOTO 400
305 IF X<8 OR X>248 THEN PLAY "s8
m20019n20":GOTO 450
310 IF X>237 THEN GJ=0:PLAY "m300
0s819n35":GOTO 500
315 IF Y>168 THEN 900
320 RETURN

```

Choque con el techo

```

400 IF X>19 AND X<35 THEN GA=1:
GOTO 420 ELSE IF X>115 AND X<
131 THEN GA=2:GOTO 420 ELSE I
F X>219 AND X<235 THEN GA=3:G
OTO 420
410 IF DR=7 THEN DR=6: K=2 ELSE
IF DR=0 THEN DR=2: K=-2 ELSE
IF DR=4 THEN DR=5:K=-2 ELSE
RETURN
415 VPOKE 6925,VPEEK(6925)+K:
RETURN

```

```

420 PLAY "m1000814n33":FOR I=1 TO
1000:NEXT:IF GA=1 OR GA=3 THE
N PT=PT+50 ELSE PT=PT+100
425 GOSUB 1000:PUT SPRITE 3,(9,11
):DR=6:RETURN

```

Choque con la pared izquierda

```

450 IF DR=4 THEN DR=7:K=-2 ELSE I
F DR=1 THEN DR=3:K=-2 ELSE DR
=6:K=-2
455 VPOKE 6924,VPEEK(6924)+K
460 RETURN

```

Choque con la pared derecha

```

500 IF DR=7 THEN DR=4:K=-2 ELSE I
F DR=6 THEN DR=5:K=2 ELSE DR=
1:K=-2
505 VPOKE 6924,VPEEK(6924)+K
510 RETURN

```

Cabezazo/patada a la pelota

```

700 IF GJ=1 THEN RETURN ELSE SPRI
TE OFF: PT=PT+1: GJ=1: GOSUB
1000:JX=VPEEK(6913):IF Y>160
THEN 720
705 IF DR=2 THEN IF RND(1)<.5 THE
N DR=4 ELSE DR=7 ELSE IF DR=6
THEN DR=0 ELSE IF DR=5 THEN D
R=0 ELSE IF DR=1 THEN DR=7 EL
SE IF DR=3 THEN DR=4
710 GOTO 750
720 IF VPEEK(6914) = 8 THEN IF X<
170 THEN DR=4:GOTO 750 ELSE D
R=1:GOTO 7
725 IF VPEEK(6914) = 4 THEN IF X>
162 THEN DR=7:GOTO 750 ELSE D
R=3:GOTO 750
750 SPRITE ON:RETURN

```



Intento de patada

```
800 JX=VPEEK(6913):IF JX+4<X THEN
VPOKE 6914,4 ELSE VPOKE 6914,
8
810 RETURN
```

Fin de juego

```
900 STRIG(0) OFF:SPRITE OFF:CLS:P
RINT "_____FIN_DE_JU
EGO":PRINT:PRINT:PRINT
910 PRINT "TUS_PUNTOS_:";PT
990 IF INKEY$<>"" THEN END ELSE
990
```

Puntuación

```
1000 FOR I=1 TO 23:PRINT:NEXT:PRIN
T "PUNTOS:";PT;CHR$(11);
1010 RETURN
```

Datos de sprites

```
10000 DATA 3,5,7,3,15,31,27,19,19,
23,6,6,6,6,6,14,128,64,192,
128,224,240,176,144,144,208,
192,192,192,192,192,224
10010 DATA 3,2,3,1,3,7,13,9,11,11,3
,1,1,1,1,1,192,128,192,128,
224,240,220,192,224,224,240,
152,142,4,0,128
10020 DATA 1,0,1,0,7,63,3,1,1,3,7,
12,56,16,0,0,224,160,224,192,
240,216,204,216,192,224,220,
224,192,64,64,192
10030 DATA 48,120,120,48,0,0,0,0
```

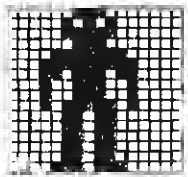
Datos de sprites pequeños

```
10100 DATA 1,2,4,8,16,32,64,128
10102 DATA 0,0,0,0,0,0,0,255
```

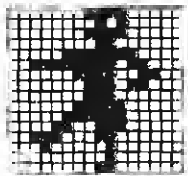
## Tabla de claves

10	= 0	110	= 362	705	= 10306
11	= 0	120	= 207	710	= 629
12	= 0	190	= 489	720	= 4836
15	= 2750	200	= 6967	725	= 4826
20	= 3995	210	= 9136	750	= 551
25	= 3996	220	= 2109	800	= 3216
30	= 3997	300	= 4073	810	= 143
35	= 3497	305	= 3496	900	= 4394
40	= 2746	310	= 3442	910	= 1383
45	= 9460	315	= 1045	990	= 1777
47	= 1851	320	= 143	1000	= 2998
50	= 5648	400	= 9460	1010	= 143
55	= 997	410	= 6676	6159	= 1343
60	= 8939	415	= 1443	10000	= 8271
61	= 2677	420	= 6520	10010	= 7438
65	= 5073	425	= 1829	10020	= 7798
70	= 1944	450	= 5567	10030	= 1273
75	= 2903	455	= 1221	10100	= 1205
80	= 2515	460	= 143	10102	= 1006
85	= 6994	500	= 5303		
95	= 1114	505	= 1221	TOTAL=	215212
100	= 1500	510	= 143		
105	= 1907	700	= 6178		

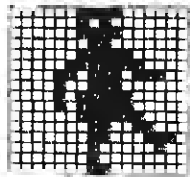
## Diseño de sprites



HOMBRE: DE PIE



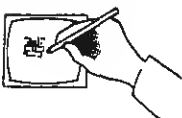
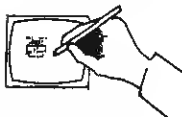
HOMBRE: PATADA  
A LA IZQUIERDA

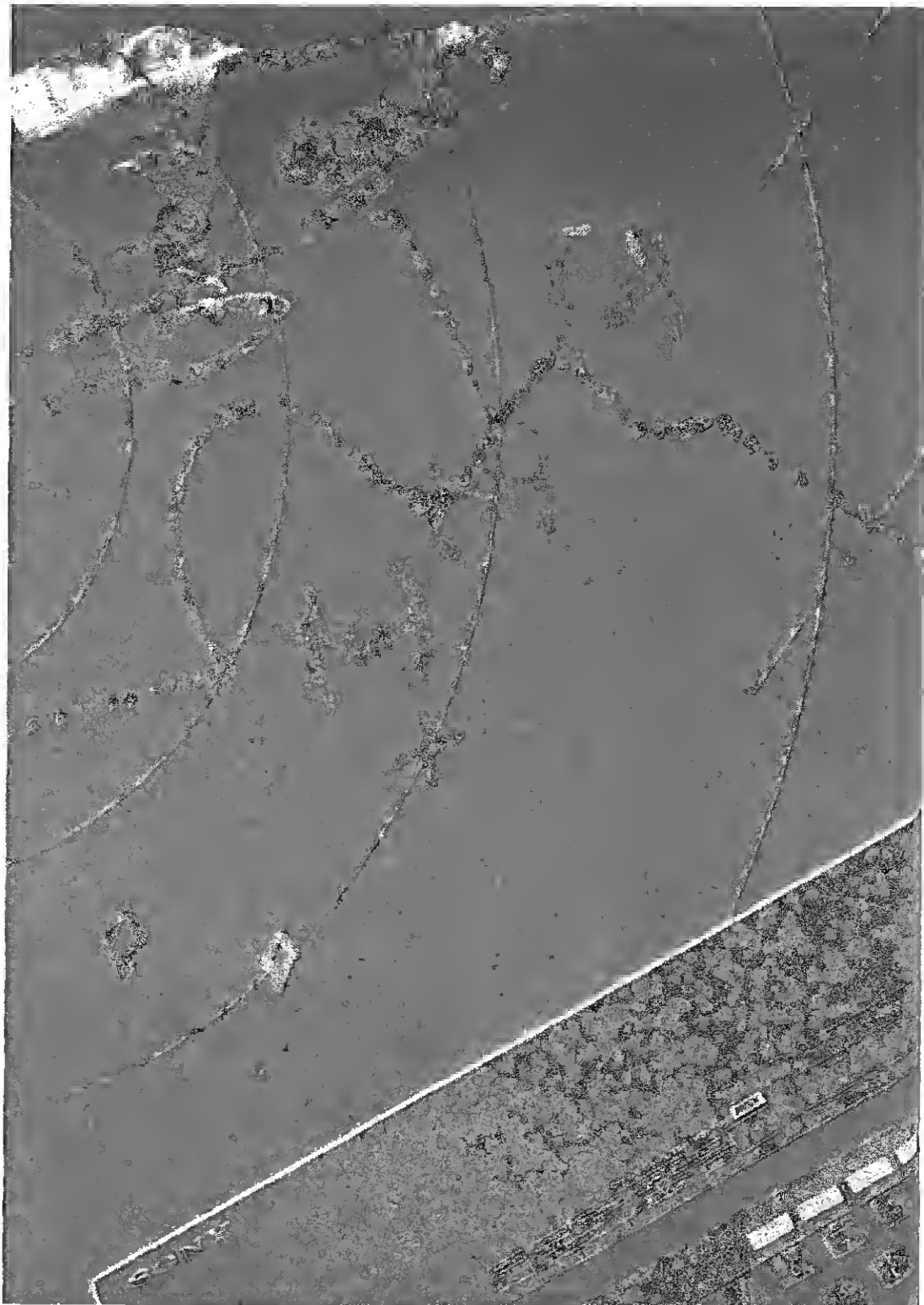


HOMBRE: PATADA  
A LA DERECHA



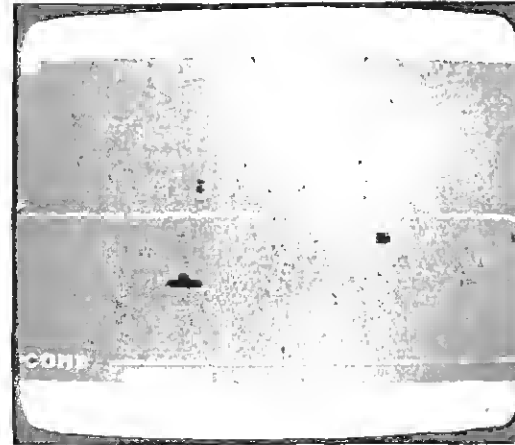
BALON





# Harrier

# 11



Tipo de juego:

JUEGO DE TIRO AL BLANCO

Misión: Lanzar cargas de profundidad desde tu avión hacia el submarino sin que explote ninguna mina. Aterrizas en el portaaviones para reponer combustible. El juego termina cuando se acabe el combustible del portaaviones.

Control:

Cursor ARRIBA	Sube o despegas.
Cursor ABAJO	Baja.
Cursor IZQUIERDA	Aterrizas.
Cursor DERECHA	Acelera.
Barra espaciadora	Lanza cargas de profundidad.

## SUGERENCIAS AL PROGRAMA

Si encuentras difícil aterrizar, en la línea 522 tienes los requisitos para un buen aterrizaje.

Añadiendo más minas se hará más difícil lanzar las cargas de profundidad sin destruirte a ti mismo.

### Variables del programa

CP Control de lanzamiento de cargas de profundidad.  
 CR Combustible que resta en tu avión.  
 TC Combustible total que queda.  
 AT Control de aterrizaje.  
 PA Control de parada.  
 RE Control de recarga de combustible.  
 CH Número de sprites que han chocado con el portaaviones.  
 SD Número de submarinos destruidos.  
 EM Explosión de una mina.

		Inicialización	
	10	REM EJECUTA EL PROGRAMA	
	11	REM SOPORTE EN CODIGO	
	12	REM MAQUINA	
	13	REM VER APENDICES	
	15	COLOR 15,4,7 : SCREEN 1,2 : KEY OFF	
	20	FOR I=1 TO 32 : READ Q : A\$=A\$+CHR\$(Q) : NEXT : SPRITE\$(0)=A\$ : PUT SPRITE 0,(120,35) : A\$=""	
	25	FOR I=1 TO 32 : READ Q : A\$=A\$+CHR\$(Q) : NEXT : SPRITE\$(1)=A\$ : A\$=""	
	30	FOR I=1 TO 8 : READ Q : A\$=A\$+CHR\$(Q) : NEXT : SPRITE\$(2)=A\$ : SPRITE\$(3)=A\$ : A\$=""	
	35	FOR I=1 TO 32 : READ Q : A\$=A\$+CHR\$(Q) : NEXT : SPRITE\$(4)=A\$ : A\$=""	
	40	FOR I=1 TO 32 : READ Q : A\$=A\$+CHR\$(Q) : NEXT : SPRITE\$(5)=A\$ : A\$=""	

		\$+CHR\$(Q) : NEXT : SPRITE\$(5)=A\$ : A\$=""	
45		FOR I=1 TO 8 : READ Q : A\$=A\$+CHR\$(Q) : NEXT : SPRITE\$(6)=A\$	
50		FOR I=1088 TO 1095 : READ Q : VPOKE I,Q : NEXT : FOR I=1152 TO 1175 : READ Q : VPOKE I,Q : NEXT : DEFUSR0=60000 : DEFUSR1=60118 : POKE 59996!,15	
55		FOR I=1 TO 10 : PRINT : NEXT : PRINT "Da una tecla para empezar"	
60		FOR I=1 TO 11 : D=RND(1) : IF INKEY\$(">") THEN 70 ELSE READ A\$ : VPOKE 6153+I,ASC(A\$) : READ N : B\$="n"+STR\$(N) : PLAY "s1m255115xB\$;" : FOR T=1 TO 500 : NEXT : NEXT	
65		FOR I=1 TO 500 : NEXT : FOR I=6154 TO 6164 : VPOKE I,32 : NEXT : RESTORE 10200 : GOTO 60	
		Preparación pantalla	
70		CLS : FOR I=6496 TO 6527 : VPOKE I,INT(RND(1)*3+144) : NEXT : FOR I=6848 TO 6879 : VPOKE I,23 : NEXT : PUT SPRITE 0,(0,16)	
72		FOR I=6848 TO 6879 : VPOKE I,23 : NEXT : PUT SPRITE 0,(0,16),8	
75		FOR I=1 TO 22 : PRINT : NEXT : PRINT "COMB:";CHR\$(11) :	
80		PUT SPRITE 4,(0,75),14 : PUT SPRITE 1,(0,140),1	
85		K1=INT(RND(1)*150+40) : PUT SPRITE 2,(K1,100),1 : K2=INT(RND(1)*50+30) : PUT SPRITE 3,(K1+K2,100),1	
90		STRIG(0) ON : ON STRIG GOSUB 900 : SPRITE ON : ON SPRITE G	

```

95  OSUB 500
    ON INTERVAL=180 GOSUB 1000 :
    INTERVAL ON : TC=30 : CR=16 :
    GOSUB 1100 : CH=100

```

Control

```

100  D=USR(D) : IF PEEK(59997!)=0
    THEN IF PEEK(59998!)=1 THEN A
    T=1
105  K1=VPEEK(6912) : IF K1>75 AND
    K1<190 THEN 700
110  GOSUB 200
115  GOSUB 250
120  IF CP=1 THEN GOSUB 300
130  IF AT=1 THEN POKE 59996!,13 E
    LSE POKE 59996!,15
190  GOTO 100

```

Movimiento del Harrier

```

200  IF PA=1 AND VPEEK(6912)>65 TH
    EN RETURN
205  IF RE=1 AND VPEEK(6912)<65 TH
    EN RE=0 : PA=0 : AT=0 : SPRIT
    E ON
215  IF AT=1 AND RE=0 THEN K1=2 :
    K2=1 : GOTO 240
220  K1=3 : K2=4
240  POKE 59997!,0 : POKE 59998!,K
    1 : POKE 59999!,K2 : D=USR1(D
    ) : RETURN

```

Movimiento del submarino

```

250  K1=RND(1) : IF K1<.5 THEN K2=
    3 ELSE IF K1<.75 THEN K2=0 EL
    SE K2=2
255  IF VPEEK(6916)<110 THEN K2=2
    ELSE IF VPEEK(6916)>160 THEN
    K2=0

```

```

260  POKE 59997!,1 : POKE 59998!,K
    2 : POKE 59999!,2 : D=USR1(D)
265  RETURN

```

Movimiento de las cargas de profundidad

```

300  POKE 59997!,6 : POKE 59998!,2
    : POKE 59999!,3 : D=USR1(D)
305  IF VPEEK(6936)>175 THEN CP=0
    : PUT SPRITE 6,(100,200)
310  RETURN

```

Choque

```

500  SPRITE OFF : IF CP=1 AND (VPE
    EK(6912) < 65 OR VPEEK(6912)>
    190 OR VPEEK(6913)>16) THEN I
    F VPEEK(6936)<90 THEN CH=6 :
    GOTO 550 ELSE IF VPEEK(6936)<
    110 THEN 650 ELSE 600
510  IF AT=0 AND CH=0 THEN CH=0 :
    GOTO 550
515  IF AT=1 THEN PA=1 : AT=0 ELSE
    IF PA=1 THEN RETURN
520  AT=0 : RE=1 : TC=TC-16+CR : I
    F TL<0 THEN 700 ELSE IF TC>0
    THEN CR=16 : GOSUB 1100
522  IF VPEEK(6913)< >5 AND VPEEK(
    6913)<>6 THEN CH=0 : GOTO 550
525  SPRITE ON : RETURN

```

Portaaviones hundido

```

550  VPOKE 6914+CH*4 , 20 : GOSUB
    1200 : FOR K1=1 TO 200 : NEXT
    : VPOKE 6930,20 : GOSUB 1200
    : FOR K1=1 TO 1000 : NEXT : G
    OTO 2000

```

Submarino bombardeado

```
600 SD=SD+1 : VPOKE 6919,15 : VPO
KE 6918,2 : GOSUB 1200 : FOR
I4=1 TO 1000 : NEXT : PUT SPR
ITE 6,(100,200) : PUT SPRITE
1,(0,140),1 : VPOKE 6918,4
640 SPRITE ON:RETURN
```

Explosión de mina

```
650 VPOKE 6938,20: VPOKE 6939,15:
GOSUB 1200: FOR K1=1 TO 1000
: NEXT : EM=1
660 GOTO 2000
```

Harrier tocado

```
700 VPOKE 6912,70 : VPOKE 6914,20:
GOSUB 1200 : FOR K1=1 TO 1000
: NEXT
710 GOTO 2000
```

Disparo de carga de profundidad

```
900 IF CP=1 THEN RETURN
905 CP=1 : I1=VPEEK(6912) : I2=VP
EEK(6913) : IF I2<8 THEN CP=0
: RETURN ELSE PUT SPRITE 6,(I2
-8,I1),1
910 RETURN
```

Depósito de combustible

```
1000 CR=CR-1 : VPOKE 6888+CR,32
1010 IF CR<0 THEN 700 ELSE RETURN
```

Recarga de combustible

```
1100 FOR I3=6888 TO 6903 : VPOKE I
3,136 : NEXT
1120 RETURN
```

Explosión

```
1200 PLAY"13m200s8n27"
1210 RETURN
```

Fin de juego

```
2000 SCREEN 1 :PRINT".....FIN_
DE_JUEGO"
2010 PRINT : PRINT
2015 PRINT"Numero_De_Subs_Destruid
os:";SD : PRINT : PRINT
2020 IF CH=0 THEN PRINT".....FUE_UN
_MAL_ATERRIZAJE" :PRINT: PRIN
T".....HAS_SIDO_DESTRUIDO!!!"
: GOTO 2100
2025 IF CH=6 THEN PRINT".....HAS_T
IRADO_UNA_CARGA.....DE_PR
OFUNDIDAD_SOBRE" : PRINT".....
..TU_PROPIO_BARCO" : GOTO
2100
2030 IF EM=1 THEN PRINT "EXPLOSION
_DE_MINA_NUCLEAR!!!" : GOTO
2100
2035 IF TC<0 OR CR<0 THEN PRINT".....
_SIN_COMBUSTIBLE!!!" : GOTO
2100
2100 IF INKEY$<>"" THEN END ELSE
2100
```

Datos de sprites

```
10000 DATA 0,0,0,0,0,192,224,241,
```

	255, 127, 127, 31, 63, 124, 0, 0, 0, 0,	
	0, 0, 0, 0, 112, 236, 242, 255, 238,	
	216, 0, 0, 0, 0	
10010	DATA 0, 0, 0, 0, 0, 2, 7, 6, 7, 127,	
	255, 255, 127, 0, 0, 0, 0, 0, 0, 0, 0,	
	128, 128, 192, 254, 255, 255, 254,	
	0, 0, 0	
10020	DATA 0, 90, 60, 126, 126, 60, 90, 0	
10030	DATA 0, 0, 0, 0, 192, 192, 240, 240,	
	240, 255, 255, 255, 255, 255, 255,	
	255, 0, 0, 0, 0, 0, 0, 0, 0, 255, 254	
	, 252, 248, 240, 224, 192	
10040	DATA 0, 4, 66, 136, 2, 64, 18, 42, 10	
	, 165, 43, 144, 40, 1, 64, 18, 128, 16	
	, 4, 32, 1, 72, 33, 146, 204, 22, 200,	
	92, 64, 9, 0, 32	
10050	DATA 40, 56, 16, 16, 56, 56, 56, 16	
	Datos de sprites pequeños	
10100	DATA 255, 171, 213, 171, 213, 171,	
	213, 255	
10102	DATA 0, 0, 4, 154, 97, 0, 0, 0	
10104	DATA 0, 0, 0, 13, 210, 32, 0, 0	
10106	DATA 0, 0, 8, 181, 66, 0, 0, 0	
	Datos de pantalla inicial	
10200	DATA S, 30, E, 32, A, 35, " ", 34, H,	
	35, A, 34, R, 33, R, 32, I, 33, E, 34, R	
	, 35	

## Tabla de claves

10	= 0	30	= 4903	65	= 3890
11	= 0	35	= 3995	70	= 6993
12	= 0	40	= 3996	72	= 2731
13	= 0	45	= 3500	75	= 2558
15	= 1236	50	= 7907	80	= 1794
20	= 5019	55	= 4252	85	= 7394
25	= 3870	60	= 11277	90	= 2947

95	= 5391	510	= 2468	2000	= 1901
100	= 4656	515	= 2917	2010	= 350
105	= 2960	520	= 5605	2015	= 3623
110	= 362	522	= 3850	2020	= 6838
115	= 410	525	= 551	2025	= 9130
120	= 986	550	= 6284	2030	= 3938
130	= 2469	600	= 6961	2035	= 3940
190	= 489	640	= 551	2100	= 1613
200	= 2072	650	= 3212	10000	= 7050
205	= 3902	660	= 603	10010	= 6280
215	= 3110	700	= 2669	10020	= 1411
220	= 850	710	= 603	10030	= 8226
240	= 3703	900	= 916	10040	= 7419
250	= 4849	905	= 5859	10050	= 1411
255	= 3840	910	= 143	10100	= 1873
260	= 3303	1000	= 1893	10102	= 1110
265	= 143	1010	= 1361	10104	= 1156
300	= 3164	1100	= 2188	10106	= 1112
305	= 2930	1120	= 143	10200	= 3662
310	= 143	1200	= 1130		
500	= 11364	1210	= 143	TOTAL=	261451

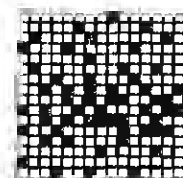
## Diseño de los sprites



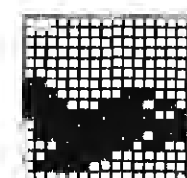
MINA



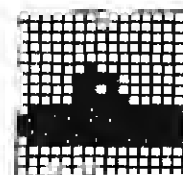
CARGA  
DE PROFUNDIDAD



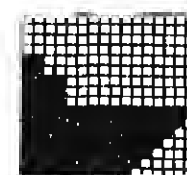
EXPLOSION



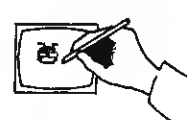
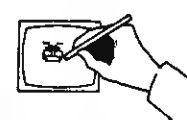
HARRIER



SUBMARINO

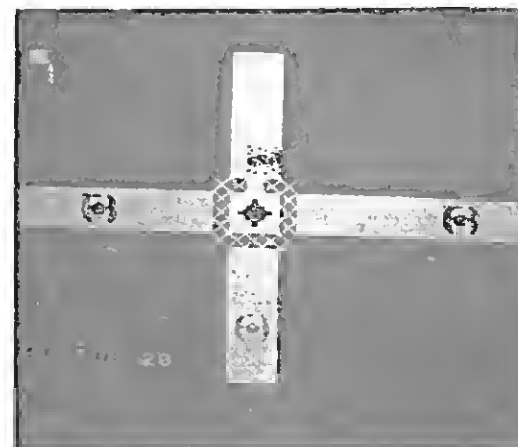


PORTAAVIONES



# Invasión de robots

# 12



Tipo de juego:

JUEGO DE DISPARO

Flotas de naves-robot suicidas están atacando tu nave de caza. Debes usar tu cañón giratorio para destruir a los robots antes de que lleguen a tu nave. Algunos robots desaparecen de repente; ten cuidado, son peores, pero valen más puntos.

Usa la tecla I para girar el cañón y la barra espaciadora para disparar.



## SUGERENCIAS AL PROGRAMA

Te darás cuenta de que la velocidad de las bombas y de los robots aumenta según juegas. Este efecto lo produce la línea 155, que fija la distancia en que se mueve cada sprite. Intenta cambiar la expresión que se "pokea" en la dirección 59999, para dar más rapidez al juego.

### Variables del programa

NG, PT Número de giros; puntos.  
D(I) Dirección del camino I.  
C Número de camino (1-4).  
DI Control de disparo.  
MS, S(I) Movimiento de sprite; control del sprite I.  
TD Todos los robots destruidos.

○		Inicialización	○
○	5	TIME=0	○
○	10	REM EJECUTA EL PROGRAMA	○
○	11	REM EN CODIGO MAQUINA	○
○	12	REM VER APENDICES	○
○	15	COLOR 15,4,7:SCREEN 1,2:CLS:KEY OFF: PRINT"___INVASION___DE___ROBOTS___":PRINT:PRINT:PRINT:PRINT	○
○	20	FOR I=1 TO 8:READ Q:A\$=A\$+CHR\$(Q):NEXT:SPRITE\$(0)=A\$:A\$=""	○
○	25	FOR I=1 TO 32:READ Q:A\$=A\$+CHR\$(Q):NEXT:SPRITE\$(1)=A\$:SPRITE\$(2)=A\$:SPRITE\$(3)=A\$:SPRITE\$(4)=A\$:A\$=""	○
○	30	FOR I=1 TO 32:READ Q:A\$=A\$+CHR\$(Q):NEXT:SPRITE\$(5)=A\$:A\$=""	○
○	35	FOR I=1 TO 32:READ Q:A\$=A\$+CHR\$(Q):NEXT:SPRITE\$(6)=A\$	○
○	40	DEFUSR=60118!	○
○	45	D(1)=2:D(2)=3:D(3)=0:D(4)=1	○
○	50	INPUT "___PULSA___RETURN___PARA___"	○

○	60	JUGAR":X\$ C=1	○
○		Preparación pantalla	○
○	100	CLS:COLOR 11,6,6	○
○	105	FOR I=6158 TO 6894 STEP 32:FOR J=0 TO 2:VPOKE I+J,255:NEXT:NEXT	○
○	110	FOR I=6464 TO 6559:VPOKE I,255:NEXT	○
○	115	FOR I=6445 TO 6449:VPOKE I,28:NEXT:FOR I=6464 TO 6528 STEP 32:VPOKE I+13,28:VPOKE I+17,28:NEXT	○
○	120	FOR I=6573 TO 6577:VPOKE I,28:NEXT	○
○	125	PUT SPRITE 5,(116,83),1:GOSUB 1000	○
○		Nueva jugada	○
○	150	TD=0:NG=NG+1:PUT SPRITE 1,(116,0),8:PUT SPRITE 2,(0,83),1:PUT SPRITE 3,(116,192),8:PUT SPRITE 4,(255,83),1	○
○	155	POKE 59999!,INT(NG/3)+1	○
○	160	GOSUB 350:STRIG(0) ON:ON STRIG GOSUB 320	○
○	170	SPRITE ON:ON SPRITE GOSUB 500	○
○	175	FOR I=1 TO 4:S(I)=1:NEXT	○
○		Control	○
○	200	IF DI=1 THEN 220	○
○	210	GOSUB 300	○
○	220	IF DI=1 THEN GOSUB 400	○
○	230	SPRITE OFF:GOSUB 450:SPRITE	○
○	ON		○
○	235	ST\$=STR\$(TIME):DG=VAL(RIGHT\$(	○

```

240 ST$,1))
    IF DG>1 OR RND(1)<.95 THEN
200
245 IF RND(1)<.5 THEN PUT SPRITE
    2,(116,200):S(2)=0:GOTO 260
250 TD=1:FOR I=1 TO 4: IF S(I)=1
    THEN TD=0
260 TD=1:FOR I=1 TO 4: IF S(I)=1 T
    HEN TD=0
265 NEXT: GOTO 200

```

Leer del teclado

```

300 KP$=INKEY$: IF KP$<>"I" AND KP
    $<>"I" THEN RETURN
305 C=C+1: IF C=5 THEN C=1
310 GOSUB 350:RETURN
320 IF DI=1 THEN RETURN
322 PLAY"119m380s10n37"
325 DI=1:ON C GOTO 330,335,340,
    345
330 PUT SPRITE 0,(120,70),1:
    RETURN
335 PUT SPRITE 0,(103,87),1:
    RETURN
340 PUT SPRITE 0,(120,103),1:
    RETURN
345 PUT SPRITE 0,(137,87),1:
    RETURN

```

Visualizar el búnker

```

350 VPOKE 6447,28:VPOKE 6509,28:V
    POKE 6575,28:VPOKE 6513,28
360 ON C GOTO 365,370,375,380
365 VPOKE 6447,255:RETURN
370 VPOKE 6509,255:RETURN
375 VPOKE 6575,255: RETURN
380 VPOKE 6513,255: RETURN

```

Disparar bomba

```

400 TP=PEEK(59999!):POKE 59997!,0
    :POKE 59999!,2
405 POKE 59998!,C-1:D=USR(D):POKE
    59999!,TP
410 ON C GOTO 415,420,425,430
415 IF VPEEK(6912)<5 THEN DI=0:PU
    T SPRITE 0,(200,200):RETURN E
    LSE RETURN
420 IF VPEEK(6913)<5 THEN DI=0: P
    UT SPRITE 0,(200,200):RETURN
    ELSE RETURN
425 IF VPEEK(6912)>187 THEN DI=0:
    PUT SPRITE 0,(200,200):RETURN
    ELSE RETURN
430 IF VPEEK(6913)>250 THEN DI=0:
    PUT SPRITE 0,(200,200):RETURN
    ELSE RETURN

```

Movimientos del enemigo

```

450 MS=MS+1: IF MS=5 THEN MS=1
455 IF TD=1 THEN 150
460 IF S(MS)=0 THEN 450
465 POKE 59997!,MS:POKE 59998!,D(
    MS):D=USR(D)
470 ON MS GOTO 475,480,485,490
475 IF VPEEK(6916)>62 THEN 600 EL
    SE RETURN
480 IF VPEEK(6921)>95 THEN 600 EL
    SE RETURN
485 IF VPEEK(6924)<111 THEN 600 E
    LSE RETURN
490 IF VPEEK(6929)<145 THEN 600 E
    LSE RETURN

```

Nave bombardeada

```

500 STRIG(0) OFF:SPRITE OFF : DI=

```

```

0: IF C=1 OR C=3 THEN PT=PT+
20:ELSE PT=PT+100
510 GOSUB 1000:S(C)=0:PUT SPRITE
C,(10,200):VPOKE 6914,24
520 PLAY "t255164r64m70s10n26n2"
530 FOR T=1 TO 150: NEXT: VPOKE
6914,0:PUT SPRITE 0,(200,200)
535 TD=1: FOR I=1 TO 4: IF S(I)=1
THEN TD =0
540 NEXT:STRIG(0) ON: SPRITE ON:
RETURN

```

Fin de juego

```

600 STRIG(0) OFF
630 CLS:SCREEN 1: PRINT "____FI
N_DE_JUEGO":PRINT:PRINT:PRINT
:PRINT:PRINT"____TUS_PUNTOS
__";PT
640 END

```

Puntuación

```

1000 FOR I=1 TO 22:PRINT: NEXT: PR
INT "PUNTOS";PT;CHR$(11);:
RETURN

```

Datos de sprites

```

10000 DATA 24,24,60,255,255,60,24,
24
10010 DATA 167,48,96,96,224,193,226
,254,231,199,195,224,96,96,48
,16,8,12,6,6,7,131,71,127,231
,227,195,7,3,3,6,4
10020 DATA 1,1,1,3,15,12,27,251,251
,27,12,15,3,1,1,1,128,128,128
,192,240,48,216,223,223,216,
48,240,192,128,128,128
10030 DATA 128,4,64,0,144,5,40,134,

```

```

67,177,68,16,68,0,128,16,4,0,
129,16,6,129,146,34,25,162,16
,65,0,8,0,129

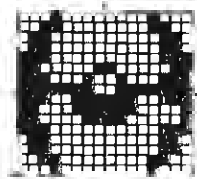
```

## Tabla de claves

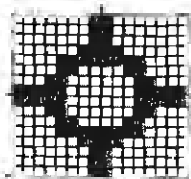
5	= 461	235	= 2960	430	= 3676
10	= 0	240	= 2330	450	= 2127
11	= 0	245	= 3553	455	= 962
12	= 0	250	= 2648	460	= 1189
15	= 4904	260	= 2648	465	= 2849
20	= 4011	265	= 784	470	= 1914
25	= 6939	300	= 3338	475	= 1756
30	= 3996	305	= 1757	480	= 1797
35	= 3484	310	= 460	485	= 1822
40	= 972	320	= 911	490	= 1875
45	= 2088	322	= 1260	500	= 5523
50	= 2534	325	= 1733	510	= 2558
60	= 325	330	= 1088	520	= 1911
100	= 582	335	= 1081	530	= 2681
105	= 3564	340	= 1121	535	= 2648
110	= 1874	345	= 1376	540	= 1513
115	= 4969	350	= 2154	600	= 766
120	= 1777	360	= 1345	630	= 5044
125	= 1395	365	= 821	640	= 129
150	= 6193	370	= 885	1000	= 3189
155	= 1888	375	= 949	10000	= 1519
160	= 1945	380	= 889	10010	= 8179
170	= 1176	400	= 3024	10020	= 8480
175	= 1499	405	= 2802	10030	= 7606
200	= 1012	410	= 1557		
210	= 207	415	= 3483		
220	= 1089	420	= 3484		
230	= 1278	425	= 3611		

TOTAL= 185927

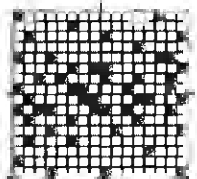
Diseño de los sprites



NAVE ENEMIGA



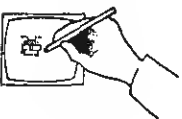
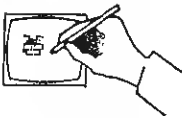
CAZA PROPIO



EXPLOSION

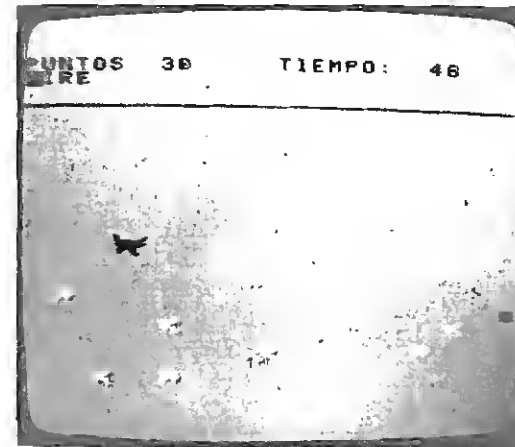


BOMBA



# Cocodrilo

# 13



Tipo de juego:

**JUEGO CON TIEMPO LIMITE**

Ayuda al cocodrilo hambriento a comer, guiándolo en dirección a las plantas, los peces exóticos y los deliciosos peces amarillos para que coma.

Se tiene que comer todos los peces amarillos y situarse en la parte de arriba de la pantalla dentro del tiempo límite.

No te olvides de que los cocodrilos no pueden estar toda la vida bajo el agua; se te puede ahogar el pobrecito. Estáte atento a la parte superior de la pantalla, donde se te indica la medida de aire que te queda.

## SUGERENCIAS AL PROGRAMA

En la línea 35 se almacena un número en la dirección 59999 para controlar la velocidad del cocodrilo; cámbialo a 2 para aumentar el nivel de dificultad, y a 4 para disminuirlo.

¿Qué pasaría si el cocodrilo tuviera más grandes los pulmones? Si aumentas el número 20 de la línea 165, el cocodrilo podrá permanecer más tiempo debajo del agua.

## Variables del programa

AI	Aire inicial.
TC	Tiempo total consumido.
AR	Aire que queda.
PN	Modelo de números del sprite 0.
R, PT	Número de inmersiones; puntuación.
RA	Control de retoma de aire.
NC	Número de cocodrilos.
TR	Tiempo que resta para llegar al otro lado.

		Inicialización	
10	REM EJECUTA EL PROGRAMA SOPOR		
11	TE		
12	REM EN CODIGO MAQUINA		
15	KEY OFF: COLOR 1,4,4:SCREEN 1		
	,2: PRINT "COCODRILLO		
	S!":PRINT:PRINT:PRINT:PRINT		
20	FOR I=1 TO 32: READ Q: A\$=A\$+		
	CHR\$(Q): NEXT: SPRITE\$(0)=A\$:		
	A\$=""		
25	FOR I=1 TO 8: READ Q: A\$=A\$+C		
	HR\$(Q): NEXT: FOR I=1 TO 4: S		
	PRITE\$(I)=A\$:NEXT:A\$=""		
30	FOR I=1 TO 32: READ Q: A\$=A\$+		
	CHR\$(Q):NEXT:SPRITE\$(5)=A\$		
35	NC=3: DEFUSR = 60000!: POKE		
	59996!,13:POKE 59999!,3:SPRIT		
	E ON :ON SPRITE GOSUB 500		
40	FOR I=60350! TO 60401!: READ		
	Q: POKE I,Q: NEXT: DEFUSR1 =		

		60350!: POKE 60346!,96: POKE	
		60347!,24: POKE 60348!,255: P	
		OKE 60349!,26	
45	FOR I=1088 TO 1280 STEP 64:FO		
	R J=0 TO 7:READ Q:VPOKE I+J,Q		
	:NEXT:NEXT		
50	PRINT"PULSA UNA TECLA PARA E		
	MPEZAR"		
55	IF INKEY\$="" THEN 55		
		Nueva inmersión	
100	CLS: IF NC=0 THEN 600		
105	FOR I=6240 TO 6271:VPOKE I,23		
	: NEXT		
110	FOR I=6274 TO 6882 STEP 32: F		
	OR J=0 TO 29:K=10*(TIME/10-IN		
	T(TIME/10)):IF RND(1)<.95 THE		
	N 140		
115	IF K<1 THEN VPOKE I+J,144:GOT		
	O 140		
120	IF K<6 THEN VPOKE I+J,136:GOT		
	O 140		
125	VPOKE I+J,152		
140	NEXT:NEXT		
145	K=K*3+80: PUT SPRITE 1,(30,K+		
	30),11:PUT SPRITE 2,(80,K+77)		
	,11:PUT SPRITE 3,(120,K+20),		
	11:PUT SPRITE 4,(200,K+60),11		
155	R=R+1		
160	PUT SPRITE 0,(0,80),1		
165	TIME=0: AI=20-R: TG=0: GOSUB		
	1000:PRINT CHR\$(11):PRINT "AI		
	RE":GOSUB 1100		
170	ON INTERVAL=15 GOSUB 650:INTE		
	RVAL ON		
180	VPOKE 8209,36:VPOKE 8210,20:V		
	POKE 8211,148:VPOKE 8212,164		
		Control	
200	D=USR(D):SPRITE ON: IF PN=0 TH		

```

205 EN PN=20 ELSE PN=0
    GOSUB 1000:VPOKE 6914,PM:IF A
    R=AI THEN 700
210 IF TR<1 THEN 600
211 PRINT TR
220 IF VPEEK(6912)<20 THEN GOSUB
    400:POKE 59996!,12:GOTO 230 E
    LSE IF VPEEK(6912)<185 THEN P
    OKE 5996!,13 ELSE POKE 59996!
    ,9
225 RA=0
230 IF VPEEK(6913)<253 THEN 240
235 IF VPEEK(6916)=200 AND VPEEK(
    6920)=200 AND VPEEK(6924)=200
    AND VPEEK(6928)=200 THEN 100
    ELSE CLS:INTERVAL OFF : FOR I
    =0 TO 5: PUT SPRITE I,(30*I,
    200):NEXT:PRINT"Has perdido, v
    arios peces!!":EL=1
236 IF EL=1 THEN PRINT:PRINT"COCC
    ODRILLOS VIVOS: ";NC-1: EL=0:
    FOR T=1 TO 2500:NEXT:NC=NC-1:
    GOTO 100
240 K1 = INT((VPEEK(6913)+14)/8)+
    32*(INT((VPEEK(6912)+4)/8))+
    6144:IF VPEEK(K1)=32 THEN 200
250 IF VPEEK(K1)=152 THEN GOSUB
    450
260 IF VPEEK(K1)=144 THEN 550
270 IF VPEEK(K1)=136 THEN GOSUB
    300
280 GOTO 200

```

Comer algas

```

300 INTERVAL OFF:PT=PT+10:VPOKE K
    1,32:GOSUB 1000
310 INTERVAL ON:RETURN

```

Toma de aire del cocodrilo

```

400 IF RA=1 THEN 420
410 TG=TG+INT(TIME/60):TIME=0:GOS

```

```

420 UB 100:RA=1
    RETURN
Comer peces exóticos
450 INTERVAL OFF:PT=PT+20:VPOKE K
    1,32:GOSUB 1000
455 PLAY"154m1200s10n54"
460 INTERVAL ON:RETURN
Comer peces amarillos
500 SPRITE OFF:I1=VPEEK(6913):IF
    I1<39 THEN 510 ELSE IF I1<89
    THEN 520 ELSE IF I1<129 THEN
    530 ELSE GOTO 540
510 IF I1>22 THEN RETURN
512 PUT SPRITE 1,(30,200): GOTO
    549
520 IF I1>72 THEN RETURN
522 PUT SPRITE 2,(80,200): GOTO
    549
530 IF I1>112 THEN RETURN
532 PUT SPRITE 3,(120,200): GOTO
    549
540 IF I1>192 THEN RETURN
542 PUT SPRITE 4,(200,200)
549 PT=PT+150:PLAY"124m160s8n27n2
    7":GOSUB 1000:RETURN

```

Choque con una mina

```

550 INTERVAL OFF:FORI=1 TO 4:COLO
    R 1,15,15:PLAY"m8000s112n30":
    FOR J=1 TO 150: NEXT: COLOR 1
    ,4,4: FOR J=1 TO 150: NEXT:
    NEXT
555 CLS:FOR I=0 TO 5:PUT SPRITE I
    ,(30*I,200):NEXT:PRINT"El coc
    odrilo chocó con minas":PRINT
    "Ha sufrido indigestión!!!"

```

○	560	PRINT:PRINT:PRINT:PRINT" _COCO DRILOS_VIVOS: _";NC-1:FOR J=1 TO 2500:NEXT	○
○	580	NC=NC-1: GOTO 100	○
○		<div>Fin de juego</div>	○
○	600	SCREEN1:PRINT" _ _ _ _ _FIN _D E_JUEGO":PRINT:PRINT:PRINT:	○
○	610	PRINT	○
○	620	PRINT" _ _ _ _Tus_puntos_son _";PT IF INKEY\$="" THEN END ELSE 620	○
○		<div>Animación del juego</div>	○
○	650	D=USR1(D): RETURN	○
○		<div>Cocodrilo ahogado</div>	○
○	700	CLS:INTERVAL OFF:FOR I=0 TO 5 :PUT SPRITE I,(30*I,200):NEXT :PRINT" _ _ _ _ _COCODRILO_AHOOGADO _ _ _"	○
○	705	PRINT:PRINT:PRINT:PRINT" _ _COC ODRILOS_VIVOS: _";NC-1	○
○	710	PLAY"11m59000s8n2": FOR T=1 T O 2500:NEXT	○
○	720	NC=NC-1: GOTO 100	○
○		<div>Tiempo/puntos</div>	○
○	1000	AR=INT(TIME/60):VPOKE 6183+AR ,32	○
○	1010	TR=50-TG-AR:PRINT CHR\$(11);"P UNTOS _";PT;TAB(19);"TIEMPO: _" ;TR	○
○	1020	RETURN	○

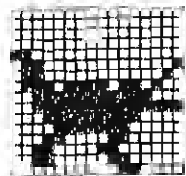
○		<div>Recarga de aire</div>	○
○	1100	FOR I=6183 TO 6183+AI: VPOKE I,160: NEXT: RETURN	○
○		<div>Datos de sprites</div>	○
○	10000	DATA 0,0,0,0,192,96,96,119,63 ,63,31,15,24,48,48,0,0,0,0,0, 0,1,2,180,248,255,240,224,96, 192,96,48	○
○	10010	DATA 128,196,110,63,63,110, 196,128	○
○	10020	DATA 0,0,0,128,192,192,96,119 ,63,63,31,15,24,48,97,97,0,0, 0,0,0,0,16,184,255,255,240, 224,96,192,128,128	○
○		<div>Datos de animación</div>	○
○	10100	DATA6,31,42,186,235,43,35,205 ,27,235,120,254,31,202,215, 235,4,43,205,44,235,35,195, 227,235,6,0,17,31,0,25,205,44 ,235,183,237,82,237,91,188, 235,123,189,194,196,235,122, 188,194,196,235,201	○
○		<div>Datos de sprites pequeños</div>	○
○	10200	DATA 146,84,185,82,181,24,16, 40	○
○	10202	DATA 0,90,60,126,126,60,90,0	○
○	10204	DATA 3,6,108,176,240,108,6,3	○
○	10206	DATA 255,255,255,255,255,255, 255,255	○



## Tabla de claves

10	= 0	211	= 314	549	= 3288
11	= 0	220	= 7159	550	= 7556
12	= 0	225	= 410	555	= 9168
15	= 4070	230	= 1726	560	= 4386
20	= 3995	235	= 15257	580	= 1374
25	= 5137	236	= 7271	600	= 2903
30	= 3483	240	= 7006	610	= 2181
35	= 4465	250	= 1778	620	= 1416
40	= 7435	260	= 1476	650	= 916
45	= 3576	270	= 1602	700	= 6512
50	= 2530	280	= 593	705	= 2983
55	= 1002	300	= 3189	710	= 2465
100	= 1101	310	= 1304	720	= 1374
105	= 1640	400	= 968	1000	= 2267
110	= 7003	410	= 3284	1010	= 4868
115	= 2170	420	= 143	1020	= 143
120	= 2155	450	= 3183	1100	= 2334
125	= 814	455	= 1304	10000	= 7116
140	= 320	460	= 1304	10010	= 1790
145	= 7496	500	= 6270	10020	= 7901
155	= 673	510	= 905	10100	= 21635
160	= 780	512	= 1366	10200	= 1579
165	= 4269	520	= 961	10202	= 1411
170	= 2876	522	= 1422	10204	= 1403
180	= 1936	530	= 1001	10206	= 1888
200	= 3166	532	= 1466		
205	= 2202	540	= 1081		
210	= 911	542	= 1016		
				TOTAL=	254509

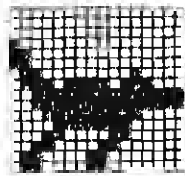
## Diseño de los sprites



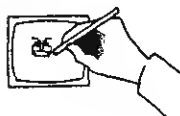
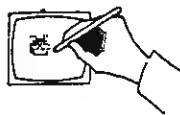
COCODRILO:  
BOCA ABIERTA



PEZ AMARILLO

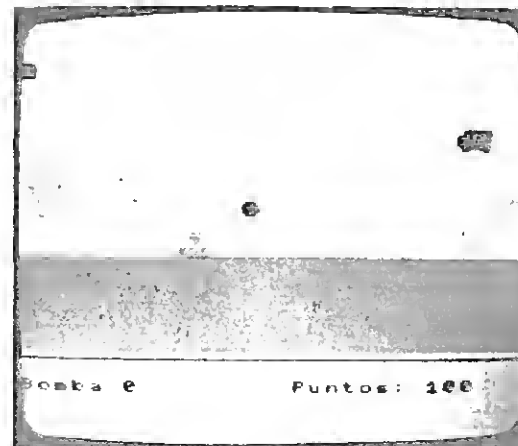


COCODRILO:  
BOCA CERRADA



# Convoy

# 14



Tipo de juego:

**JUEGO CON TIEMPO LIMITE**

Eres un saboteador y tu misión es detener un convoy de camiones enemigos. Se trata de colocar unas bombas de relojería bajo la carretera, moviendo tu barca bajo un río subterráneo.

El problema es que tus camiones (pintados de rojo) deben poder pasar, y no puedes controlar la mecha de tus bombas.

Para poner una bomba, o arreglar la carretera destruida, pulsa la barra espaciadora, pero ten cuidado con los camiones que caen.

## SUGERENCIAS AL PROGRAMA

Los camiones que caen parecerían más grandes si salpicaran mucho e hicieran bastante ruido al golpear el agua. Estos efectos se podrían incluir en la rutina "Caída de camión" en las líneas 550-580.

## Variables del programa

UC	Ultimo camión.
P	Puntuación.
DM	Distancia máxima entre camiones.
C(I)	Colores de los sprites.
NA, NF	Número aleatorio; semilla del número aleatorio.
LM, CM	Longitud mecha; control mecha.
VC	Velocidad camiones.
CJ	Columna jugador.
PB	Posición bomba.
CP	Control posición bomba.
UM	Ultimo camión movil.

## Inicialización

```

10 REM EJECUTA EL PROGRAMA
11 REM EN CODIGO MAQUINA
12 REM VER APENDICES
15 COLOR 1,7,7:SCREEN 1,2:KEY OFF:PRINT"_____CONVOY_":
FOR I=1 TO 9:PRINT:NEXT
20 FOR I=1 TO 32: READ Q: A$=A$+
CHR$(Q): NEXT: SPRITE$(0)=A$:
A$=""
25 FOR I=1 TO 32: READ Q: A$=A$+
CHR$(Q): NEXT: FOR I= 1 TO 4
:SPRITE$(I)=A$:NEXT
30 PUT SPRITE 1,(20,50),1:PUT SPRITE 2,(80,50),8:PUT SPRITE 3
,(160,50),1:PUT SPRITE 4,(220
,50),8
35 DEFUSR0=60000!:DEFUSR1=60118!
:POKE 59996!,10
40 PRINT"__PULSA UNA TECLA PARA EL

```

```
MPEZAR_"
IF INKEY$="" THEN 50
VC=2
FOR I=1088 TO 1095 STEP 2: VP
OKE I,170: VPOKE I+1,85: NEXT
: FOR I=1152 TO 1159: VPOKE I
,255: NEXT: VPOKE 8209,231: V
POKE 8210,71
```

Preparación pantalla

```

100  CLS: FOR I=6336 TO 6463: VPOKE
    E I,136: NEXT: FOR I=6592 TO
    6815:VPOKE I,144:NEXT
105  FOR I=6816 TO 6847:VPOKE I,23
    :NEXT
110  FOR I=1 TO 23: PRINT: NEXT: P
    RINT"Bomb:";CHR$(11);: GOSUB
    1000
120  PUT SPRITE 1,(10,200):PUT SPR
    ITE 2,(40,200):PUT SPRITE 3,(
    70,200):PUT SPRITE 4,(100,200
    ):UC=1:DM=25:GOSUB 400
130  PUT SPRITE 0,(120,95),11
140  STRIG(0) ON : ON STRIG GOSUB
    950
150  NF = 10001*(TIME/60-INT(TIME/
    60))
160  ON SPRITE GOSUB 750

```

Control

```

200 POKE 59999!,5: D=USR0(D)
205 GOSUB 900: IF CM=0 AND NA<.2 T
HEN GOSUB 900: LM=INT(NA*5+1):
CM=1: TIME=0
207 GOSUB 1100
210 LT=LT+1: IF LT=5 THEN LT=1
212 IF VPEEK(6912+4*LT)=200 THEN
210 ELSE POKE 59997!,LT: POKE
59998!,1: POKE 59999!,VC: D=USR
1(D): IF (VPEEK(6913+4*LT)<2+VC
)AND C(LT)=1 THEN 800

```

```

213 IF VPEEK(6913+4*LT)<2+VC THEN
215 PUT SPRITE LT, (30*LT-20,200)
C=INT((VPEEK(6913+4*LT)+8)/8)
: IF VPEEK(6368+C)=32 AND VPEEK(6367+C)=32 AND VPEEK(6369+C)=32 THEN GOSUB 550
220 IF CM=1 AND CF=0 AND INT(TIME/600)>LM THEN 700
225 IF CP=1 AND INT(TIME/600)>LM THEN GOSUB 450
230 GOSUB 900: IF NA<.05 THEN GOSUB 400
240 GOTO 200

```

Reparar carretera

```

350 PLAY"s8m1000s14n53":IF CJ<3 THEN 380
352 IF CJ>28 THEN 392
355 FOR I1=6336 TO 6432 STEP 32:FOR I2=CJ-2 TO CJ+2
360 IF VPEEK(I1+I2)=32 THEN VPOKE I1+I2,136
365 NEXT: NEXT: RETURN
380 FOR I1=6336 TO 6432 STEP 32:FOR I2=0 TO CJ
385 IF VPEEK(I1+I2)=32 THEN VPOKE I1+I2,136
390 NEXT: NEXT: RETURN
392 FOR I1=6336 TO 6432 STEP 32:FOR I2=29 TO 31
394 IF VPEEK(I1+I2)=32 THEN VPOKE I1+I2,136
396 NEXT: NEXT: RETURN

```

Nuevo camión

```

400 J=0 : FOR I=1 TO 4: IF VPEEK(6912+4*I)=200 THEN J=1
405 NEXT
410 IF J=0 THEN RETURN
415 XP=VPEEK(6913+4*UC):IF 255-XP<DM THEN RETURN

```

```

420 UC=J:GOSUB 900:IF NA<.4 THEN C(UC)=1 ELSE C(UC)=8
425 PUT SPRITE UC, (255,34),C(UC)
430 RETURN

```

Explosión de bomba

```

450 PLAY"s8m200164n30n28n26n24":CM=0:VPOKE 6464+PB,32:CF=0:IF PB=31 THEN 470
455 IF PB=0 THEN 480
460 FOR J=6336 TO 6432 STEP 32:FOR K=PB-1 TO PB+1
465 VPOKE J+K,32:NEXT:NEXT:RETURN
470 FOR J=6336 TO 6432 STEP 32:FOR K=29 TO 31
475 VPOKE J+K,32:NEXT:NEXT:RETURN
480 FOR J=6336 TO 6432 STEP 32:FOR K=0 TO 2
485 VPOKE J+K,32:NEXT:NEXT:RETURN

```

Colocar bomba

```

500 CF=1:PB=CJ:VPOKE 6464+CJ,2
510 PLAY"154m1200s10n35"
520 RETURN

```

Caída de camión

```

550 SPRITE ON:POKE 59997!,LT:POKE 59998!,2:POKE 59999!,3
555 D=USR1(D):IF VPEEK(6912+4*LT)<100 THEN 555
560 IF C(LT)=8 THEN 800
565 PT=PT+50:VPOKE 6912+4*LT,200:VPOKE 6913+4*LT,LT*30-20
570 IF PT<300 THEN VC=3 ELSE IF PT<500 THEN VC=4 ELSE IF PT<700 THEN VC=5 ELSE IF PT<900 THEN VC=6 ELSE IF PT<1400 THEN VC=7 ELSE VC=8

```

```

580 GOSUB 400: SPRITE OFF: GOSUB
1000: RETURN

Jugador destruido

700 FOR I=1 TO 40: GOSUB 900: A$="n
"+STR$(INT(10*NA+1)): PLAY "I6
4m350s8xa$;": VPOKE 14336+INT(
60*NA), NA*255: NEXT
740 GOTO 800

Hombre aplastado

750 CLS: PRINT: PRINT "____ACABAS_D
E_SER_APLASTADO": FOR I=1 TO
2500: NEXT

Fin de juego

800 STRIG(0) OFF: SCREEN 1: PRINT "
____FIN_DE_JUEGO": FOR I=1 TO
9: PRINT: NEXT: PRINT "____TUS
PUNTOS SON"; PT
810 IF INKEY$="" THEN END ELSE
810

Número aleatorio

900 NA=(9999*NA+NF)MOD 2997!: NA=N
A/2997!
910 IF NA<.5 THEN NF=NA*10000+1
920 RETURN

Espacio pulsado

950 CJ=INT((VPEEK(6913)+8)/8)
955 IF VPEEK(6432+CJ)<>32 AND CP=
0 THEN GOSUB 500: RETURN
960 IF VPEEK(6432+CJ)=32 THEN GOS

```

```

965 UB 350: RETURN
RETURN

Puntuación

1000 FOR I=1 TO 22: PRINT: NEXT: PRIN
T TAB(16) CHR$(10) "Puntos: "; P
T: CHR$(11);
1010 RETURN

Tiempo restante

1100 I=FL-INT(TIME/60): IF I<0 THEN
I=0
1110 IF CM=0 THEN VPOKE 6888, 32: RE
TURN ELSE VPOKE 6888, 48+1:
RETURN

Datos de sprites

10000 DATA 1, 19, 17, 15, 7, 3, 1, 3, 2, 6,
255, 255, 255, 127, 63, 31, 128, 200
, 136, 240, 224, 192, 128, 192, 64,
96, 255, 255, 255, 254, 252, 248
10010 DATA 0, 15, 23, 18, 114, 149, 181,
255, 248, 231, 95, 63, 56, 0, 0, 0, 0,
255, 69, 21, 75, 101, 21, 255, 0, 255
, 254, 254, 14, 0, 0, 0

```

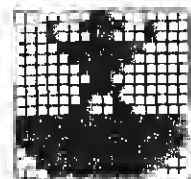
### Tabla de claves

10	= 0	50	= 997	150	= 2684
11	= 0	60	= 416	160	= 750
12	= 0	70	= 6615	200	= 1540
15	= 4324	100	= 4236	205	= 5284
20	= 3995	105	= 1768	207	= 238
25	= 4580	110	= 3182	210	= 2131
30	= 4196	120	= 6006	212	= 11038
35	= 2781	130	= 899	213	= 4249
40	= 2563	140	= 1687	215	= 8831

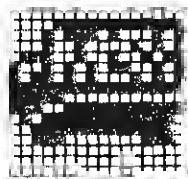
220	= 3475	425	= 1443	740	= 391
225	= 2884	430	= 143	750	= 4136
230	= 1693	450	= 5538	800	= 6598
240	= 593	455	= 1026	810	= 1349
350	= 2323	460	= 2824	900	= 3287
352	= 949	465	= 1256	910	= 2115
355	= 2973	470	= 2076	920	= 143
360	= 2590	475	= 1256	950	= 2104
365	= 525	480	= 1988	955	= 3177
380	= 2274	485	= 1256	960	= 2056
385	= 2590	500	= 1921	965	= 143
390	= 525	510	= 1305	1000	= 4161
392	= 2199	520	= 143	1010	= 143
394	= 2590	550	= 2898	1100	= 2857
396	= 525	555	= 2924	1110	= 2988
400	= 3627	560	= 965	10000	= 8898
405	= 131	565	= 4315	10010	= 7900
410	= 838	570	= 9693		
415	= 3424	580	= 1476		
420	= 3532	700	= 8332		

TOTAL= 230474

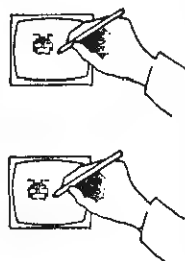
## Diseño de los sprites



HOMBRE EN LA BARCA



CAMION



# El gorila matemático

# 15



Tipo de juego:

EDUCATIVO

Godofredo está ansioso de subir la colina y encontrar el gorila matemático. Cuando respondas una pregunta correctamente, la roca que cae no golpeará a Godofredo, llevándolo al principio, y éste se acercará al gorila.

Hay dos niveles de dificultad y las preguntas son sobre suma y multiplicación. Cuando tengas la respuesta pulsa <RETURN>.

## SUGERENCIAS AL PROGRAMA

Quizá puedas animar al gorila mientras cae la roca, o que estreche la mano a Godofredo cuando se encuentren, o un grito agudo cuando la roca golpea a Godofredo...

### Variables del programa

C	Centinela.
ND	Nivel de dificultad.
P	Pregunta (1-2).
MR	Modelos de rocas (8 ó 12).
N1, N2	Números a sumar/multiplicar.
NC	Nuevo centinela.
RJ	Respuesta del jugador.
VR	Velocidad de la roca.
M\$	Mensaje.

		Inicialización	
5	COLOR 15,4,7:SCREEN 1,3:KEY OFF:ON INTERVAL=1 GOSUB 7:INTERVAL ON		
6	GOTO 5		
7	D=RND(1):RETURN		
8	FOR I=6145 TO 6148:VPOKE I,23:NEXT:VPOKE 6144,24:VPOKE 6175,25:VPOKE 6880,26:VPOKE 6911,27:FOR I=6176 TO 6848 STEP 32:VPOKE I,22:VPOKE I+31,22:NEXT:FOR I=6881 TO 6910:VPOKE I,23:NEXT		
10	PRINT "GODFREDO":PRINT:PRINT "MULTIPLICA":PRINT:PRINT "MUESTRA":PRINT:PRINT "MULTIPLICA":FOR I=1 TO 9:PRINT:NEXT		
20	FOR I=1 TO 32:READ Q:A\$=A\$+CHR\$(Q):NEXT:SPRITE\$(0)=A\$:A\$=""		
25	FOR I=1 TO 32:READ Q:A\$=A\$+CHR\$(Q):NEXT:SPRITE\$(1)=A\$:A\$=""		

30	FOR I=1 TO 8:READ Q:A\$=A\$+CHR\$(Q):NEXT:SPRITE\$(2)=A\$:A\$=""		
35	FOR I=1 TO 8:READ Q:A\$=A\$+CHR\$(Q):NEXT:SPRITE\$(3)=A\$		
37	FOR I=1088 TO 1119:READ Q:VPOKE I,Q:NEXT		
40	PUT SPRITE 1,(116,30),1		
45	INPUT "Quieres sumar(s) o multiplicar(m) numeros";X\$		
50	IF X\$="m" OR X\$="M" THEN P=2:GOTO 60		
55	IF X\$="s" OR X\$="S" THEN P=1 ELSE 45		
60	PRINT:INPUT "NIVEL DE DIFICULTAD(1/2)";X\$:IF X\$<"1" OR X\$>"2" THEN 60		
65	ND=VAL(X\$)		
70	CLS:FOR I=6374 TO 6397:VPOKE I,138:NEXT		
75	VPOKE 8209,228:PUT SPRITE 1,(92,23):PUT SPRITE 0,(0,104),9		
80	FOR I=0 TO 5:VPOKE 6176+33*I,136:NEXT:FOR I=0 TO 6:VPOKE 6495+31*I,137:NEXT		
85	FOR I=6208 TO 6368 STEP 32:FOR J=0 TO K:VPOKE I+J,139:NEXT:K=K+1:NEXT		
90	K=0:FOR I=6527 TO 6719 STEP 32:FOR J=-K TO 0:VPOKE I+J,139:NEXT:K=K+1:NEXT		
95	INTERVAL OFF:SPRITE ON:ON SPRITE GOSUB 700		
		Imprimir pregunta	
100	CD=0:NC=0:MR=8:RJ=0:PUT SPRITE 2,(0,1)		
102	M\$="Cual es la respuesta?":GOSUB 1000		
105	IF P<>1 THEN 120		
110	IF ND=1 THEN N1=INT(RND(1)*10+1):N2=INT(RND(1)*10+1) ELSE		



```

N1=INT(RND(1)*20+1):N2=INT(RND(1)*20+1)
115 K=43:GOTO 140
120 IF ND=1 THEN N1=INT(RND(1)*5+1):N2=INT(RND(1)*5+1) ELSE N1=INT(RND(1)*10+1):N2=INT(RND(1)*10+1)
125 K=120
140 FOR I=1 TO 20:PRINT:NEXT:PRINT TAB(6);:PRINT USING"##";N1;:PRINT TAB(9);CHR$(K);TAB(11);:PRINT USING"##";N2;:PRINT TAB(14);"=_?_";CHR$(11);
150 C=0

```

Control

```

300 GOSUB 400:IF NC=1 THEN 100
310 GOTO 500
390 GOTO 300

```

Movimiento de roca

```

400 IF ND=1 THEN VR=2 ELSE VR=3
405 K1=VPEEK(6921):K2=VPEEK(6920):IF K1/4<>INT(K1/4) THEN IF MR=8 THEN MR=12 ELSE MR=8
410 VPOKE 6922,MR:IF K1<49 AND K2<46 THEN VPOKE 6921,K1+VR:VPOKE 6920,K2+VR:RETURN
420 IF K2<50 AND K1<240 THEN VPOKE 6921,K1+VR:IF K1>62 AND K1<125 THEN VPOKE 6916,0:RETURN ELSE VPOKE 6916,23:RETURN
425 IF K2<80 THEN VPOKE 6920,K2+VR:RETURN
430 IF K2<125 THEN VPOKE 6920,K2+VR:VPOKE 6921,K1-VR:RETURN
440 IF K1<5 THEN NC=1:RETURN ELSE VPOKE 6921,K1-VR:RETURN

```

Leer

```

500 X$=INKEY$:IF X$="" THEN 300
505 IF (X$<"0" OR X$>"9") AND X$<>CHR$(13) THEN 300
510 IF X$=CHR$(13) THEN 600
515 CD=CD+1:IF CD>3 THEN 600 ELSE RJ=RJ*10+VAL(X$):VPOKE 6801+CD,48+VAL(X$):GOTO 300

```

Comprobar respuesta

```

600 IF CD=0 THEN 300 ELSE M$="_____"
:IF P<>1 THEN 650
605 IF N1+N2<>RJ THEN 300
610 GOTO 660
650 IF N1*N2<>RJ THEN 300
660 IF NC=1 THEN 300 ELSE C=1:SPRITE OFF:K1=VPEEK(6920):K2=VPEEK(6921):IF K1<50 THEN VPOKE 6912,23:IF K2<200 THEN VPOKE 6913,K2 ELSE VPOKE 6913,200:PUT SPRITE 2,(100,200):NC=1 ELSE EL=1
661 IF EL=1 THEN IF K2>195 THEN VPOKE 6913,165 ELSE VPOKE 6913,K2:EL=0
665 SPRITE ON:GOTO 300

```

Choque de sprites

```

700 SPRITE OFF:IF C=1 THEN GOSUB 1100:RETURN
705 IF VPEEK(6912)<70 AND VPEEK(6913)<115 THEN 900
710 FOR I2=1 TO 10:VPOKE 6912,VPEEK(6912)-10:PLAY"m380s8164n56":FOR I1=1 TO 100:NEXT:VPOKE

```

```

720 6912,VPEEK(6912)+10:PLAY "m70
    0164s8n50":NEXT
    C=0:PUT SPRITE 0,(0,104):PUT
    SPRITE X,(0,0):NC=1:SPRITE ON
    :RETURN

```

Fin de juego

```

900 SCREEN 1:FOR I=1 TO 10:PRINT:
    NEXT:PRINT "VIVA!!!!_has_en
    contrado_a1":PRINT "Gorila_ma
    tematico"
910 PLAY "m255s118o5edco4bo5co4b5
    agao5co4bo5d11c"
990 END

```

Imprimir mensaje

```

1000 FOR I=1 TO 23:PRINT:NEXT:PRIN
    T"___";M$;CHR$(11);
1010 RETURN

```

Salto del jugador

```

1100 PLAY "164n60":NC=1:C=0:K3=VPE
    EK(6913):K4=VPEEK(6912):IF K4
    >50 THEN IF K3<165 THEN VPOKE
    6913,K3+20:SPRITE ON:RETURN
    ELSE VPOKE 6912,23:SPRITE ON:
    RETURN
1110 VPOKE 6913,K3-30:SPRITE ON:
    RETURN

```

Datos de sprites

```

10000 DATA 1,2,1,1,7,15,13,8,9,11,3
    ,3,3,3,6,14,192,160,192,64,

```

```

10010 DATA 3,5,31,59,113,248,237,
    199,135,135,207,30,28,12,6,30
    ,128,64,240,184,28,62,110,198
    ,194,194,230,240,112,96,192,
    240
10020 DATA 96,240,06,0,0,0,0,0
10030 DATA 64,224,224,64,0,0,0,0

```

Datos de sprites pequeños

```

10100 DATA 128,64,160,80,168,84,170
    ,85
10102 DATA 1,2,5,10,21,42,85,170
10104 DATA 255,170,85,170,85,170,85
    ,170
10106 DATA 170,85,170,85,170,85,170
    ,85

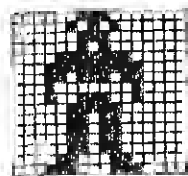
```

## Tabla de claves

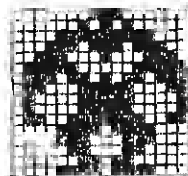
5	= 4269	85	= 4522	425	= 1931
6	= 394	90	= 5026	430	= 2987
7	= 1018	95	= 2776	440	= 2750
8	= 11539	100	= 3025	500	= 1569
10	= 11865	102	= 3826	505	= 3352
20	= 4256	105	= 1091	510	= 1370
25	= 4255	110	= 9892	515	= 6721
30	= 3916	115	= 978	600	= 8827
35	= 3398	120	= 9978	605	= 1589
37	= 2172	125	= 453	610	= 539
40	= 827	140	= 9122	650	= 1591
45	= 9238	150	= 325	660	= 13012
50	= 2651	300	= 1288	661	= 3663
55	= 2453	310	= 633	665	= 851
60	= 5294	390	= 433	700	= 1638
65	= 1018	400	= 1895	705	= 2719
70	= 2282	405	= 6714	710	= 9895
75	= 2375	410	= 4867	720	= 3184
80	= 4875	420	= 6687	900	= 8765

910 = 4201	1110 = 1365	10100 = 1642
990 = 129	10000 = 6990	10102 = 1279
1000 = 3322	10010 = 9496	10104 = 1721
1010 = 143	10020 = 1153	10106 = 1640
1100 = 10033	10030 = 1279	TOTAL= 272972

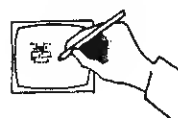
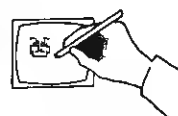
## Diseño de los sprites



HOMBRE



GORILA



ROCA:  
POSICION 1



ROCA:  
POSICION 2

# Rescate aéreo 16



Tipo de juego:

JUEGO DE SIMULACION

Un rascacielos necesita, desesperadamente, unos paquetes especiales, los cuales debes recoger con tu helicóptero de los edificios más bajos y llevarlos al rascacielos. En éste no puedes aterrizar, ya que no hay helipuerto, y podría producir un accidente.

Debes recoger cuatro paquetes. Tus puntos son:

Objetivo alcanzado	100.
Edificio alcanzado	50
Recogida satisfactoria	10

Cuanto más rápido trabajes, mejor será tu puntuación. Utiliza la barra espaciadora para soltar el paquete.

## SUGERENCIAS AL PROGRAMA

Juega con los números asignados a J y las sentencias RND (1) de las líneas 200-220 para aumentar o disminuir el vaivén del helicóptero y su velocidad.

Un buen ejercicio de programación con gráficos sería mejorar los edificios y el paisaje, y ¿qué me dices de poner ventanas y coches aparcados?

Para alargar el juego puedes añadir más paquetes y hacer que se entreguen en un orden determinado

### Variables del programa

P	Puntuación.
PD	Paquetes dejados.
VH	Velocidad helicóptero.
DR	Direcciones (diagonal ascendente o descendente).
C	Cargado.
M1, M2	Número de paquetes cogidos de cada edificio.

## Inicialización

```

5      REM RESCATE AEREO
10     REM EJECUTA EL PROGRAMA
11     REM SOPORTE EN CODIGO MAQUINA
12     REM VER APENDICES
30     COLOR 11,1,1:SCREEN 2,2
35     FOR I=1 TO 75:READ X1,Y1,X2,Y
      2:LINE (X1,Y1)-(X2,Y2),11:
      NEXT
40     PAINT(100,100):PAINT(170,130)
      :PAINT(85,135):PAINT(70,130):
      PAINT(40,100): PAINT(20,100):
      PAINT(210,170):PAINT(140,180)
      :PAINT(40,160)
50     FOR I=1 TO 32: READ Q: A$=A$+
      CHR$(Q): NEXT: SPRITE$(0)=A$:
      PUT SPRITE 0,(240,0),10:A$=""
55     FOR I=1 TO 32: READ Q: A$=A$+
      CHR$(Q):NEXT:FOR I=1 TO 4:SPR
      ITE$(I)=A$:NEXT:A$="":PUT SPR

```

```

ITE 1, (226,70),14:PUT SPRITE
2, (241,70),14: PUT SPRITE 3, (
226,100),14:PUT SPRITE 4, (241
,100),14
FOR I=1 TO 32: READ Q: A$=A$+
CHR$(Q): NEXT: SPRITE$(5)=A$:
A$=""
FOR I=1 TO 32: READ Q: A$=A$+
CHR$(Q): NEXT: SPRITE$(6)=A$:
A$=""
FOR I=1 TO 32: READ Q: A$=A$+
CHR$(Q):NEXT:SPRITE$(7)=A$
DEFUSR=60000!:DEFUSR1=60118!:
POKE 59996!,15:POKE 59997!,0:
FD=4:DR=2:VH=1:TIME=0:C=1
STRIG(0) ON : ON STRIG GOSUB
800

```

Control

```

100     POKE 59999!,VH:D=USR(D)
110     GOSUB 200
120     GOSUB 300
121     C=1
190     GOTO 100

```

Movimiento del helicóptero

```

200      IF RND(1)<.7 THEN K=DR: J=1:
        GOTO 215
205      J=4: IF RND(1)<.5 THEN K=1:
        GOTO 215
210      K=3
215      POKE 59999!,J: POKE 59998!,K:
        D=USR1 ( D )
220      RETURN
300      K=VPEEK(6918): IF K<4 OR K>191
        THEN VPOKE 6912,4: RETURN

```

Comprobar posición

```

305 IF K>160 THEN 500
310 IF K<30 THEN RETURN
315 X=VPEEK(6913):IF X>209 THEN V
POKE 6913,209:IF K>160 THEN
500 ELSE RETURN
320 IF X<150 THEN 330 ELSE IF K <
80 THEN RETURN ELSE IF X>200
THEN 500 ELSE IF X<165 THEN R
ETURN ELSE IF C=1 THEN 500 EL
SE M2=M2+1:GOTO 400
330 IF K>140 THEN 500 ELSE IF X <
60 THEN 340 ELSE IF K>50 THEN
RETURN ELSE 500
340 IF K<60 THEN RETURN ELSE IF K
>100 THEN 500 ELSE IF K>80 TH
EN RETURN ELSE IF C=1 THEN
500 ELSE M1=M1+1: GOTO 400
390 RETURN

```

Recoger paquete

```

400 C=1:PUT SPRITE PD,(235,130):P
UT SPRITE 6,(233,160),15:DR=0
: PT=PT+10: VPOKE 6912,VPEEK(
6912)-2
410 IF M1>2 OR M2>2 THEN 700
440 RETURN

```

Choque

```

500 PUT SPRITE 6,(100,200): PUT
SPRITE 7,(233,160),15: VPOKE
6914,20:FOR I=1 TO 30:PLAY"16
01000s14n50":NEXT:FOR I=1 TO
2500: NEXT: PUT SPRITE 7,(100
,200)
550 GOTO 900

```

Edificio incorrecto

```

700 SCREEN 1:PRINT "DEBES_COGER_D
OS_PAQUETES_":PRINT "DE_CADA_
EDIFICIO": FOR I=1 TO 3000:
NEXT
710 PRINT "Intentalo_otra_vez":IF
INKEY$="" THEN END ELSE 710

```

Dejar paquete

```

800 IF C=0 THEN END
805 PUT SPRITE 6,(100,200):PLAY"1
8m1000s14n40":C=0:DR=2:EL=0
806 I1=VPEEK(6913):IF I1<150 THEN
820 ELSE IF I1<170 THEN PUT S
PRITE PD,(I1,160):PD=PD-1:GOT
O 850 ELSE IF I1<200 THEN PUT
SPRITE PD,(I1,90):PD=PD-1:GOT
O 850 ELSE PUT SPRITE PD,(I1,
160):PD=PD-1: GOTO 850
820 IF I1>140 THEN PUT SPRITE PD,
(I1,90):PD=PD-1:GOTO 850 ELSE
IF I1<60 THEN 840 ELSE IF I1<
72 THEN PUT SPRITE PD,(I1,145
):PD=PD-1:GOTO 850 ELSE EL=1
821 IF EL=1 THEN IF I1<100 OR I1>
106 THEN PUT SPRITE PD,(I1,32
):PT=PT+50: PD=PD-1: GOTO 850
830 PUT SPRITE PD,(I1,32): PT=PT+
100:PLAY "m1100s813n50":PD=PD
-1:GOTO 850
840 IF I1<22 OR I1>55 THEN PUT SP
RITE PD,(I1,110):PD=PD-1:GOTO
850 ELSE PUT SPRITE PD,(I1,70)
:PD=PD-1
850 IF PD=0 THEN FOR I1=1 TO 2000
:NEXT:GOTO 900
860 RETURN

```

Fin de juego

```

900 SCREEN 1:PRINT"FIN_
DE_JUEGO":PRINT:PRINT:PRINT
910 PRINT "PUNTOS_";PT
915 IF PDK>0 THEN 940 ELSE K=150-
TIME: IF K<0 THEN K=0
920 PRINT:PRINT:PRINT"MAS_BONOS_P
OR_LLEVAR_TODOS":PRINT"LOS_PA
QUETES:";INT(8*K):PRINT:PRINT
"ESTO_TE_DA_UN_TOTAL_DE_";PT+
INT(8*K)
940 IF INKEY$="" THEN END ELSE
940

```

Datos del dibujo

```

10000 DATA 100,14,100,18,100,14,148
,41,100,14,72,41,100,18,145,
44,100,18,75,42,120,68,148,41
,120,68,72,41,120,68,120,160,
72,41,91,145,148,41,140,140
10002 DATA 0,100,120,160,0,112,110,
168,88,190,110,168,110,190,
140,160,140,160,195,190,120,
160,153,127,149,150,176,165,
186,169,225,190,186,169,225,
131,176,165,225,118
10004 DATA 98,136,109,142,98,136,
100,150,100,150,108,142,109,
142,110,155,102,43,120,43,102
,43,113,50,102,43,109,36,113,
50,120,43,113,50,109,36,109,
36,120,43
10006 DATA 180,70,159,91,180,70,214
,90,180,73,161,91,180,73,213,
91,180,70,180,73,159,91,194,
109,159,91,150,150,194,109,
177,164,214,90,192,149,194,
109,215,90

```

```

10008 DATA 79,139,75,120,75,120,87,
126,75,120,84,111,78,137,61,
128,61,128,59,122,59,122,76,
130,59,122,82,100,60,130,46,
83,46,83,21,70,46,83,59,70
10010 DATA 59,70,35,57,35,57,21,70,
21,70,40,119,40,117,16,105,16
,105,10,93,10,93,35,105,10,93
,24,78,0,190,88,190,110,190,
195,190,59,70,70,112
10012 DATA 143,103,152,107,152,107,
150,125,150,125,140,134,142,
116,152,107,205,116,219,102,
219,102,212,97,219,102,209,
128,209,128,195,141,198,172,
225,186,198,172,204,160,204,
160,225,171,204,160,217
10014 DATA 146,217,146,225,150,225,
190,225,30,225,30,255,30

```

Datos de sprites

```

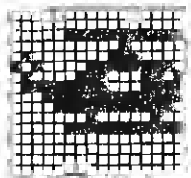
10100 DATA 0,0,255,4,14,127,177,241
,255,127,32,255,0,0,0,0,0,0,
224,0,2,133,194,254,248,240,
128,224,0,0,0,0
10110 DATA 1,2,4,8,21,26,29,30,31,
31,31,15,7,3,1,0,128,64,32,16
,8,24,40,200,136,136,136,144,
160,192,128,0
10120 DATA 0,4,66,136,2,64,18,42,10
,165,43,144,40,1,64,18,128,16
,4,32,1,72,33,146,204,22,200,
92,64,9,0,32
10130 DATA 255,255,255,227,221,190,
190,190,190,190,221,227,255,
255,255,255,255,255,187,
183,175,159,175,18
10140 DATA 192,160,198,170,170,198,
0,0,0,0,0,255,0,0,0,0,0,198
,170,170,166,2,2,10,4,0,255,0
,0,0,0

```

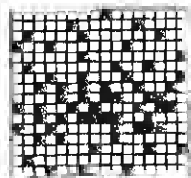
## Tabla de claves

5	= 0	220	= 143	850	= 2658
10	= 0	300	= 3084	860	= 143
11	= 0	305	= 1131	900	= 2663
12	= 0	310	= 868	910	= 1325
30	= 719	315	= 4179	915	= 3569
35	= 3584	320	= 8482	920	= 9965
40	= 7597	330	= 4059	940	= 1481
50	= 5263	340	= 6748	10000	= 13270
55	= 12463	390	= 143	10002	= 16393
60	= 3996	400	= 6160	10004	= 14772
65	= 3997	410	= 1616	10006	= 14760
67	= 3481	440	= 143	10008	= 12789
70	= 6544	500	= 8529	10010	= 12062
80	= 1496	550	= 523	10012	= 22370
100	= 1648	700	= 5948	10014	= 2786
110	= 362	710	= 3921	10100	= 7539
120	= 207	800	= 824	10110	= 7438
121	= 325	805	= 3300	10120	= 7419
190	= 489	806	= 15211	10130	= 11545
200	= 3030	820	= 10348	10140	= 6674
205	= 2867	821	= 6348		
210	= 337	830	= 4989		
215	= 2498	840	= 6786		
				TOTAL=	336007

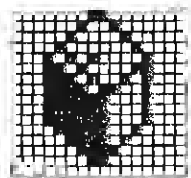
## Diseño de los sprites



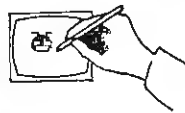
HELICOPTEROS



EXPLOSION



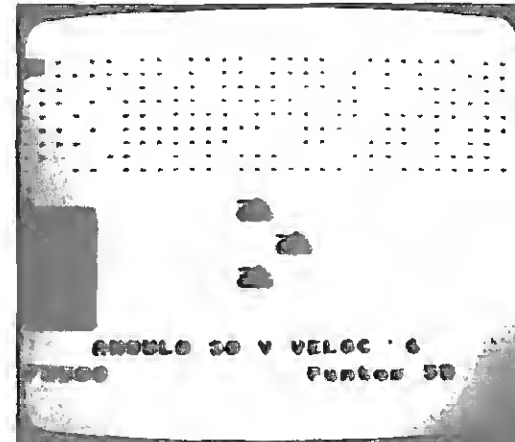
PAQUETE





# Emboscada de tanques

# 17



Tipo de juego:

JUEGO DE SIMULACION

Se ha preparado una emboscada a un escuadrón de tanques enemigos. Tu tanque está situado en un risco, desde donde disparas al enemigo que viene por la derecha. Con las teclas de 0 a 9 eliges el ángulo y velocidad de disparo; después pulsa la barra espaciadora para disparar. También puedes lanzar una bengala para poder ver los tanques enemigos pulsando sólo la barra espaciadora.

Cada tanque abatido te suma 50 puntos en el marcador. El juego termina cuando un tanque alcanza la base del risco.

## SUGERENCIAS AL PROGRAMA

Creando otros sprites y modificando la rutina "movimiento de tanques enemigos" se puede añadir más tanques.

La dificultad del juego la determina, por otra parte, la distancia entre los tanques, así que si se modifican las líneas 216 a 222 se puede obtener efectos interesantes.

## Variables del programa

NP, PT	Número de pantalla; puntuación.
M\$	Mensaje.
OP	¿Otra pantalla?
TM	Ultimo tanque movido.
F	¿Fuego?
A, V	Angulo; velocidad.
T	Tiempo.
XI, YI	Coordenadas iniciales de la bomba.
X, Y	Coordenadas calculadas.

		Inicialización	
	10	REM EJECUTA EL PROGRAMA	
	11	REM SOPORTE EN CODIGO	
	12	REM MAQUINA	
	15	SCREEN 1,2: KEY OFF: PRINT TAB(5)"EMBOSCADA_DE_TANQUES": PRINT: PRINT: PRINT: PRINT	
	20	FOR I=1 TO 32: READ Q: A\$=A\$+CHR\$(Q): NEXT: SPRITE\$(0)=A\$: A\$=""	
	25	FOR I=1 TO 32: READ Q: A\$=A\$+CHR\$(Q): NEXT: SPRITE\$(1)=A\$: SPRITE\$(2)=A\$: SPRITE\$(3)=A\$: SPRITE\$(4)=A\$: A\$=""	
	30	FOR I=1 TO 8: READ Q: A\$=A\$+CHR\$(Q): NEXT: SPRITE\$(5)=A\$: A\$=""	
	35	FOR I=1 TO 8: READ Q: A\$=A\$+CHR\$(Q): NEXT: SPRITE\$(6)=A\$	
	40	DEFUSR=60118!: POKE 59998!,1	

	45	INPUT"PULSA_'RETURN'_PARA_EMP	
		EZAR";X\$	
	50	CLS: COLOR 14,1,1	
	55	FOR I=6496 TO 6752 STEP 32: F	
		OR J=0 TO 5: VPOKE I+J,255:	
		NEXT: NEXT:FOR I=6784 TO 6815	
		: VPOKE I,23: NEXT	
	60	FOR I=1 TO 21: PRINT: NEXT: P	
		RINT"____ANGULO____Y_VELOC.":	
		PRINT CHR\$(11);: M\$="LISTO____	
		____": GOSUB 1000	
	65	FOR I=6144 TO 6431: IF RND(1)	
		<.7 THEN VPOKE I,46	
	67	NEXT: FOR I=36 TO 375: VPOKE	
		I,0: NEXT:VPOKE 369,1	
		Nuevo juego	
	70	PUT SPRITE 0,(2,74),10: TM=1:	
		F=0: K1=35*(TIME/10-INT(TIME/	
		10))+60: PUT SPRITE 1,(235,K1	
		),1	
	71	PUT SPRITE 2,(255,K1+20),1:	
		PUT SPRITE 3,(235,K1+40),1:	
		PUT SPRITE 4,(255,K1+55),1	
	75	OP=0: SPRITE ON: ON SPRITE GO	
		SUB 300	
		Control	
	80	IF F=0 THEN 95	
	85	GOSUB 150	
	95	GOSUB 200: IF OP=1 THEN NP=NP	
		+1: GOTO 70	
	100	X\$=INKEY\$: IF X\$="" OR F<>0	
		THEN 80	
	105	IF X\$=" " THEN M\$="BENGALA____	
		____": GOSUB 1000: F=2: A=	
		.628: V=9: GOSUB 150: GOTO 80	
	110	IF X\$<"0" OR X\$>"9" THEN 100	

```

115 K1=VAL(X$): VPOKE 6829,K1+48
117 X$=INKEY$: IF X$="" THEN GOSU
B 200: GOTO 117
120 IF X$<"0" OR X$>"9" THEN 117
125 A=10*K1+VAL(X$): VPOKE 6830,
VAL(X$)+48: A=A*.01745
130 X$=INKEY$: IF X$="" THEN GOSU
B 200: GOTO 130
135 IF X$<"0" OR X$>"9" THEN 130
140 V=VAL(X$): VPOKE 6841,V+48
145 F=0: X$=INKEY$: IF X$="" THEN
GOSUB 200: GOTO 145
147 IF X$="_" THEN M$="FUEGO,
" : GOSUB 1000: F=1: GOS
UB 150: GOTO 80
148 GOTO 145

```

Movimiento de bomba

```

150 T=T+3: XI=V*COS(A)/2.1: YI=V*
SIN(A)/2.1
152 IF T<>3 AND T<>0 THEN 155
153 IF F=0 THEN RETURN ELSE PUT
SPRITE 5, (20,55),15: PLAY"m80
s1164n30n20"
155 X=INT(20+T*XI)
160 Y=INT(55-T*YI+.049*T*T)
165 IF X<0 OR X>247 OR Y>154 THEN
GOSUB 400: RETURN
170 IF Y<0 THEN PUT SPRITE 5, (100
,210) ELSE PUT SPRITE 5, (X,Y)
,15
175 IF F<>2 THEN 185
180 IF T>=30 THEN 250
185 RETURN

```

Movimiento de tanques enemigos

```

200 SPRITE OFF: TM=TM+1: IF TM=5
THEN TM=1

```

```

210 IF VPEEK(6912+TM*4)>200 THEN
RETURN
215 POKE 59997!,TM: ON NP GOTO
216,217,217,219,219,221,222
216 K=1: GOTO 225
217 K=2: GOTO 225
219 K=3: GOTO 225
221 K=4: GOTO 225
222 K=5
225 POKE 59999!,K: D=USR(D)
230 IF VPEEK(6913+TM*4)<41 THEN
600
235 SPRITE ON: RETURN

```

Lanzallamas

```

250 PUT SPRITE 5, (100,210): FOR I
=1 TO 5: COLOR 1,15,15: FOR J
=1 TO 70: NEXT: COLOR 14,1,1:
FOR J=1 TO 70: NEXT: NEXT
255 M$="LISTO,
" : GOSUB
1000: F=0: T=0: VPOKE 6829,32
: VPOKE 6830,32: VPOKE 6841,
32
270 RETURN

```

Explosión tanque enemigo

```

300 IF F<>1 THEN RETURN ELSE VPOK
E 6934,24: COLOR 1,15,15: PLA
Y"m13000s118n40n35n40n35": FO
R I1=1 TO 200: NEXT: COLOR 14
,1,1
305 I2=0: FOR I1=1 TO 4: IXD=VPEE
K(6913+I1*4)-VPEEK(6933): IYD
=VPEEK(6912+I1*4)-VPEEK(6932)
: IF IXD<0 THEN IXD=-IXD
IF IYD<0 THEN IYD=-IYD
310

```

```

315 IF VPEEK(6933)+6>=VPEEK(6913+
    11*4) AND IXD<=16 AND VPEEK(
    6932)+ 6>= VPEEK(6912+11*4)
    AND IYD<=14 THEN I2=I1
320 NEXT: VPOKE 6914+4*I2,24: FOR
    I1=1 TO 200: NEXT
325 P=P+50: VPOKE 6914+4*I2,4*I2:
    VPOKE 6934,20: PUT SPRITE 5,
    (100,210): PUT SPRITE I2,(12*
    25-25,210): F=0: T=0: M$="LIS
    TO.....": GOSUB 1000
330 IF VPEEK(6916)>200 AND VPEEK(
    6920)>200 AND VPEEK(6924)>200
    AND VPEEK(6928)>200 THEN OP=1
350 VPOKE 6829,32: VPOKE 6830,32:
    VPOKE 6841,32: RETURN

```

Explosión de bomba

```

400 VPOKE 6934,24: T=0:F=0: M$="L
    ISTO.....": GOSUB 1000
405 PLAY"m80164s8n10n12": VPOKE
    6829,32: VPOKE 6830,32: VPOKE
    6841,32: FOR I=1 TO 400: NEXT
410 VPOKE 6934,20: PUT SPRITE 5,(
    100,210)
450 RETURN

```

Fin de juego

```

600 SCREEN 1: PRINT TAB(8)"FIN_DE
    L_JUEGO": PRINT: PRINT: PRINT
    : PRINT
610 PRINT" Tus_puntos_son: ";F: PRI
    NT
620 PRINT"Has_superado";NP;:PRINT
    "oleadas":PRINT"de_tanques"
630 END

```

Programa/mensaje

```

1000 FOR I=1 TO 23: PRINT: NEXT: P
    RINT M$; TAB(17);"Puntos";F;
    CHR$(11);: RETURN

```

Datos de sprites

```

10000 DATA 0,0,63,63,63,63,127,0,
    127,200,135,200,127,0,,1,7,
    156,240,192,248,254,0,254,19,
    225,19,254,0,0,0
10010 DATA 0,0,255,1,7,31,127,0,127
    ,199,146,199,127,0,0,0,64,
    248,248,248,252,254,0,254,227
    ,73,227,254,0,0,0
10020 DATA 0,0,24,60,60,24,0,0
10030 DATA 128,4,64,0,144,5,40,134,
    67,177,68,16,68,0,128,16,4,0,
    129,16,6,129,146,34,25,162,16
    ,65,0,8,0,129

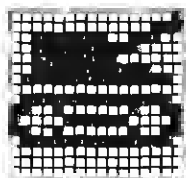
```

## Tabla de claves

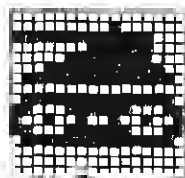
10	= 0	70	= 6283	140	= 1927
11	= 0	71	= 4595	145	= 2984
12	= 0	75	= 1473	147	= 4540
15	= 4105	80	= 809	148	= 538
20	= 3995	85	= 310	150	= 4746
25	= 6939	95	= 2651	152	= 1980
30	= 4012	100	= 2467	153	= 3437
35	= 3500	105	= 6083	155	= 1605
40	= 1767	110	= 1814	160	= 2787
45	= 2603	115	= 2004	165	= 2905
50	= 578	117	= 2501	170	= 2888
55	= 5731	120	= 1814	175	= 1144
60	= 6851	125	= 4264	180	= 1246
65	= 2746	130	= 2535	185	= 143
67	= 2223	135	= 1844	200	= 2645

210	= 2251	270	= 143	450	= 143
215	= 4060	300	= 6683	600	= 2695
216	= 1025	305	= 9158	610	= 2134
217	= 1025	310	= 1848	620	= 4360
219	= 1029	315	= 9139	630	= 129
221	= 1029	320	= 2518	1000	= 3901
222	= 337	325	= 10256	10000	= 7566
225	= 1544	330	= 6189	10010	= 7899
230	= 2094	350	= 1875	10020	= 1158
235	= 551	400	= 3273	10030	= 7606
250	= 5620	405	= 4720		
255	= 4941	410	= 1424	TOTAL=	242335

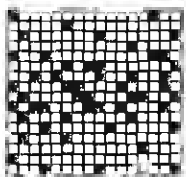
## Diseño de los sprites



TANQUE PROPIO



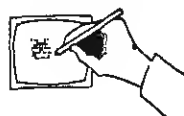
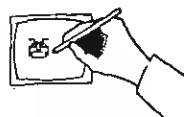
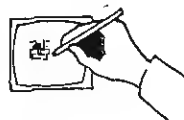
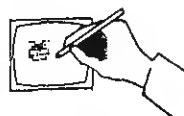
TANQUE ENEMIGO



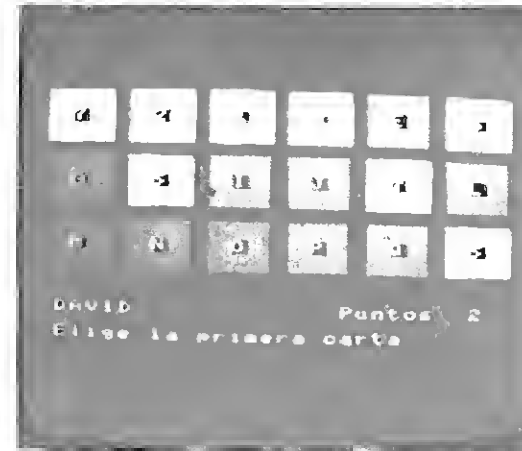
EXPLOSION



BOMBA



# Concentración 18



Tipo de juego:

JUEGO MEMORISTICO

Desafia a tu oponente a una batalla mental. Elige dos cartas entre dieciocho; si ambas tienen el mismo número, en el reverso, tienes un punto; si no, se vuelven otra vez boca abajo y se le da una oportunidad a tu contrario.

¡Que gane el que mejor memoria tenga!

## SUGERENCIAS AL PROGRAMA

Para facilitar el juego puedes usar un rango de números menor. Si, por el contrario, quieres hacerlo más difícil, en vez de tener dos grupos de números, del 1 al 9, pon tres grupos de números del 1 al 6 y haz que cada jugador elija tres cartas.

La presentación en pantalla quedaría mejor poniendo más colores a las cartas.

## Variables del programa

J\$(1), J\$(2) Nombres de los jugadores.  
P(1), P(2) Puntuaciones.  
N(I, J) Números de las cartas.  
JC Jugador en acción.  
CL Carta a levantar.  
UC Ultima carta.

		Inicialización	
10	SCREEN 1: COLOR 15,4,4: KEY OFF: PRINT TAB(8) "CONCENTRACION"		
15	DIM N(18,2)		
20	FOR I= 1072 TO 1143: READ Q: VPOKE I,Q: NEXT: FOR I= 1152 TO 1223: READ Q: VPOKE I,Q: NEXT: VPOKE 8208,164: VPOKE 8209,164: VPOKE 8210,228: VPOKE 8211,228		
25	FOR I=1 TO 5: PRINT : NEXT: PRINT "Nombre_del_jugador_1?"		
35	X\$=INKEY\$: IF X\$="" THEN D=RD(1): GOTO 35 ELSE IF X\$=CHR\$(13) THEN J\$(1)="RICARDO": GOTO 50 ELSE J\$(1)=X\$		
40	L=LEN(J\$(1)): BEEP: VPOKE 6403+L,ASC(X\$): IF L> 15 THEN 50		
45	X\$=INKEY\$: IF X\$="" THEN D=RD(1): GOTO 45 ELSE IF X\$=CHR\$(13) THEN 50 ELSE J\$(1)=J\$(1)+X\$: GOTO 40		

50	FOR I=1 TO 5: PRINT : NEXT: PRINT "Nombre_del_jugador_2?"		
60	X\$=INKEY\$: IF X\$="" THEN 60 ELSE IF X\$=CHR\$(13) THEN J\$(2)="JULIO": GOTO 75 ELSE J\$(2)=X\$		
65	L=LEN(J\$(2)): BEEP: VPOKE 6595+L,ASC(X\$): IF L>15 THEN 75		
70	X\$=INKEY\$: IF X\$="" THEN 70 ELSE IF X\$=CHR\$(13) THEN 75 ELSE J\$(2)		
75	FOR I=1 TO 5: PRINT : NEXT: PRINT "Vale ";J\$(1);" -": PRINT "Tu_primer_carta_PULSA UNA_TECLA"		
80	IF INKEY\$="" THEN 80		
		Preparación pantalla	
100	CLS: FOR CL=1 TO 18: GOSUB 400: NEXT		
140	JC=1: GOSUB 600: I=0: J=0		
150	I=I+1: IF I= 10 THEN I=1: J=J+1: IF J=2 THEN 200		
155	K=INT(RND(1)*18+1): IF N(K,1)=0 THEN N(K,1)=1 ELSE 155		
160	GOTO 150		
		Editor	
200	PRINT CHR\$(11);: FOR I=1 TO 20: PRINT : NEXT: PRINT "Eligela_primer_carta"		
205	X\$=INKEY\$: IF X\$="" OR ((X\$<"A" OR X\$>"R") AND (X\$<"a" OR X\$>"r")) THEN 205		
210	IF X\$>"R" THEN CL= ASC(X\$)-96 ELSE CL=ASC(X\$)-64		
215	IF N(CL,2)=1 THEN 205 ELSE GOSUB 300: UC=CL		

```

220 PRINT "Ahora_la_segunda_"
225 X$=INKEY$: IF X$="" OR ((X$<"
A" OR X$>"R") AND (X$<"a" OR
X$>"r")) THEN 225
230 IF X$>"R" THEN I=ASC(X$)-96
ELSE I=ASC(X$)-64
235 IF I=CL THEN 225 ELSE CL=I:
IF N(CL,2)=1 THEN 225 ELSE GO
SUB 300
240 IF N(CL,1)=N(UC,1) THEN GOSUB
500: GOTO 250
245 PLAY"14m61680s10n2": PRINT"No
Latinaste_": FOR T=1 TO 2000:
NEXT: GOSUB 400: CL=UC: GOSU
B 400
250 IF JC=1 THEN JC=2 ELSE JC=1
255 PRINT CHR$(11);: FOR I=1 TO
19: PRINT: NEXT: FOR I=1 TO
3: PRINT "_____
_____":NEXT:PRINT "_____
_____";
260 GOSUB 600: GOTO 200

```

Levantat carta

```

300 PLAY"s8m1000120n20r30n35": K=
INT((CL-1)/6)+1: ON K GOTO
305,310,315
305 AD=6172+5*CL: GOTO 320
310 AD=6332+5*(CL-6): GOTO 320
315 AD=6492+5*(CL-12)
320 VPOKE AD,144: VPOKE AD+1,148:
VPOKE AD+2,148: VPOKE AD+3,
148: VPOKE AD+4,146: VPOKE AD
+32,150: VPOKE AD+64,150: VPO
KE AD+96,150: VPOKE AD+128,
145: VPOKE AD+129,149
321 VPOKE AD+130,149: VPOKE AD+
131,149: VPOKE AD+132,147: VP
OKE AD+100,151: VPOKE AD+68,
151: VPOKE AD+36,151
325 FOR I= AD+33 TO AD+97 STEP 32

```

```

: FOR J=0 TO 2: VPOKE I+J,152
: NEXT: NEXT: VPOKE AD+66,N(
CL,1)+48
RETURN

```

Volver cartas

```

400 PLAY"S8L50M1000N35N20":K=INT(
(CL-1)/6)+1: ON K GOTO 405,
410,415
405 AD=6172+5*CL: GOTO 420
410 AD=6332+5*(CL-6): GOTO 420
415 AD=6492+5*(CL-12)
420 VPOKE AD,134: VPOKE AD+1,138:
VPOKE AD+2,138: VPOKE AD+3,
138: VPOKE AD+4,136: VPOKE AD
+32,140: VPOKE AD+64,140: VPO
KE AD+96,140: VPOKE AD+128,
135: VPOKE AD+129,139
421 VPOKE AD+130,139: VPOKE AD+
131,139: VPOKE AD+132,137: VP
OKE AD+100,141: VPOKE AD+68,
141: VPOKE AD+36,141
425 FOR I= AD+33 TO AD+97 STEP 32
: FOR J=0 TO 2: VPOKE I+J,142
: NEXT: NEXT: VPOKE AD+66,64+
CL
490 RETURN

```

Acertaste

```

500 P(JC)=P(JC)+1
510 N(UC,2)=1: N(CL,2)=1
520 K1=204: PLAY"M2000L10S0N35N40
": GOSUB 580
521 K1=193: PLAY"110S8M1000N30N45
": GOSUB 580: K1=32: GOSUB
580
530 IF P(1)+P(2)=9 THEN 900
550 RETURN

```



```

580 FOR I=6144 TO 6175: VPOKE I,
    K1: NEXT: FOR I=6176 TO 6656
    STEP 32: VPOKE I,K1: FOR J=1
    TO 50: NEXT: PLAY "SBL64M500N
    25": VPOKE I+31,K1: NEXT: FOR
    I= 6656 TO 6687: VPOKE I,K1:
    NEXT
585 RETURN

```

Nombre/puntuación

```

600 PRINT CHR$(11):: FOR I=1 TO
    18: PRINT : NEXT : PRINT J$(J
    C)":_ " TAB(18)"Puntos:";P(JC)
690 RETURN

```

Fin de juego

```

900 CLS: PRINT "El ganador es_":
    IF P(1)>P(2) THEN PRINT J$(1)
    : PRINT "con una puntuación d
    e_";P(1) ELSE PRINT J$(2): P
    RINT "con una puntuación de_
    ";P(2)
910 PLAY"v10o414a18gel8eg12f11a",
    "o314a18eg18ge12f11f","o514c1
    8gel8ge12f11c"
920 FOR I=1 TO 3000: NEXT: PRINT:
    PRINT : PRINT"Malas suerte,_";
    : IF P(1)<P(2) THEN PRINT J$(
    1): PRINT "tus puntos son_";
    P(1) ELSE PRINT J$(2): PRINT
    "tus puntos son_";P(2)
930 PLAY "v15140s8m60000n2r40n1r3
    0n4r20n1r30n2r10n5r10n4r10n3r
    2013n1"
990 END

```

Datos de sprites pequeños

```

10000 DATA 0,0,0,0,10,5,10,5
10010 DATA 10,5,10,5,0,0,0,0
10020 DATA 0,0,0,0,160,64,160,64
10030 DATA 160,64,160,64,0,0,0,0
10040 DATA 0,0,0,0,170,85,170,85
10050 DATA 170,85,170,85,0,0,0,0
10060 DATA 10,5,10,5,10,5,10,5
10070 DATA 160,64,160,64,160,64,160
    ,64
10080 DATA 170,85,170,85,170,85,170
    ,85
10090 DATA 0,0,0,0,15,15,15,15
10100 DATA 15,15,15,15,0,0,0,0
10110 DATA 0,0,0,0,240,240,240,240
10120 DATA 240,240,240,240,0,0,0,0
10130 DATA 0,0,0,0,255,255,255,255
10140 DATA 255,255,255,255,0,0,0,0
10150 DATA 15,15,15,15,15,15,15,15
10160 DATA 240,240,240,240,240,240,
    240,240
10170 DATA 255,255,255,255,255,255,
    255,255

```

## Tabla de claves

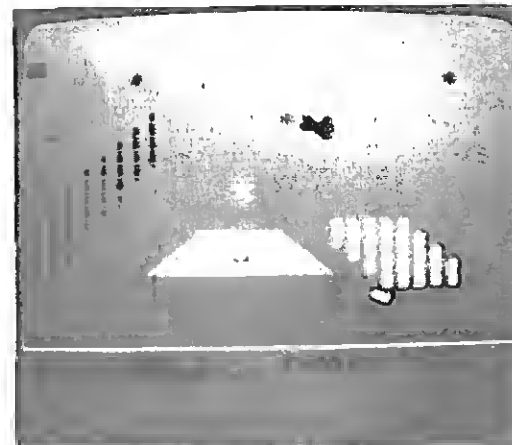
10	= 2981	100	= 1609	240	= 2639
15	= 404	140	= 1537	245	= 6223
20	= 7430	150	= 3623	250	= 1831
25	= 3580	155	= 4256	255	= 6476
35	= 7931	160	= 539	260	= 914
40	= 3766	200	= 4576	300	= 5209
45	= 7038	205	= 5522	305	= 1698
50	= 3582	210	= 4090	310	= 2270
60	= 6062	215	= 2423	315	= 1589
65	= 4015	220	= 1975	320	= 11194
70	= 5547	225	= 5510	321	= 6761
75	= 6034	230	= 3900	325	= 5972
80	= 1027	235	= 3811	390	= 143

400	= 5173	585	= 143	10060	= 1150
405	= 1798	600	= 4626	10070	= 1628
410	= 2366	690	= 143	10080	= 1640
415	= 1589	900	= 11759	10090	= 1134
420	= 11134	910	= 6550	10100	= 1150
421	= 7022	920	= 11912	10110	= 1372
425	= 5712	930	= 4938	10120	= 1404
490	= 143	990	= 129	10130	= 1384
500	= 1157	10000	= 1025	10140	= 1416
510	= 1395	10010	= 1017	10150	= 1404
520	= 2370	10020	= 1264	10160	= 1864
521	= 3325	10030	= 1272	10170	= 1888
530	= 1443	10040	= 1272		
550	= 143	10050	= 1280	TOTAL=	275051
580	= 9830				



# Acróbata aéreo

# 19



Tipo de juego:

JUEGO DE SIMULACION

Un grupo de acróbatas tratan de saltar de una avioneta a una cama elástica. Para saltar, pulsa la barra espaciadora, pero, ¡controla la velocidad del viento y la altura!, ya que pierdes puntos si caes al suelo o a la cerca de pinchos, y tu hombre muere si cae en los cables de alta tensión.

Puedes modificar la velocidad y el efecto del viento en la trayectoria del hombre cambiando los valores de las variables VV y T en la línea 310.

ND	Nivel de dificultad.
VV	Velocidad del viento.
T	Tiempo.
V	¿Vuelo?
NF, NH	Número de fallos; número de hombres.
M\$	Mensaje.
X, Y	Coordenadas del hombre.
NP, P	Número de pantalla; puntuación.
XI, YI	Coordenadas iniciales.

```

5      REM EJECUTA EL PROGRAMA
6      REM SOPORTE EN CODIGO
7      REM MAQUINA
10     SCREEN 1,2: KEY OFF: CLS: PR
      NT TAB(7)"ACROBATA_LAERED": PR
      INT: PRINT: PRINT
15     FOR I=1 TO 9: READ Q: A$=A$+
      CHR$(Q): NEXT: SFRITE$(0)=A$:
      A$=""
20     FOR I=1 TO 32: READ Q: A$=A$+
      CHR$(Q): NEXT: SFRITE$(1)=A$
25     FOR I=1088 TO 1095: READ Q: V
      POKE I,Q: NEXT: FOR I=1152 TO
      1191: READ Q: VPOKE I,Q: NEXT
26     FOR I=1216 TO 1223: READ Q: V
      POKE I,Q: NEXT: FOR I=1280 TO
      1287: READ Q: VPOKE I,Q:
      NEXT
30     DEFUSR=60118!: POKE 59997!,1:
      POKE 59998!,1
40     ON STRIG GOSUB 900
50     NH=3: P=100

```

```

100  STRIG(0) ON:CLS:COLOR 1,5,5
105  IF NH=0 THEN 600
110  FOR I=6816 TO 6847:VPOKE I,
    23:NEXT
115  PRINT:PRINT"_____"CHR$(160)
    "_____"CHR$(160)
    :PRINT:PRINT:PRINT"_____"
    "_____"CHR$(160):PRINT"_"CHR
    $(160)"____!":PRINT"____!!
    !":PRINT"____!!!!"
120  PRINT"__!!!!":PRINT"__!!!!":
    PRINT"__!!!!":PRINT"!!!!":
    PRINT"!!!!_____"CHR$(146)CHR
    $(146)CHR$(146)CHR$(146)"___"
    CHR$(148)CHR$(148)CHR$(148)CH
    R$(148)CHR$(148)
125  PRINT"!!!_____"CHR$(144)CHR$
    (136)CHR$(136)CHR$(136)CHR$(
    136)CHR$(145)
127  PRINT"__"CHR$(147)CHR$(148)CH
    R$(147)CHR$(147)CHR$(147)CHR$
    (148):PRINT"!!!_____"CHR$(144)
    CHR$(136)CHR$(136)CHR$(136)C
    HR$(152)CHR$(136)CHR$(136)CHR
    $(136)CHR$(136)CHR$(145);
129  PRINT"__"CHR$(147)CHR$(148)"_
    _"CHR$(147)CHR$(148):PRINT"!!
    _____"CHR$(144);:FOR I=1 TO
    9:PRINT CHR$(136);:NEXT:PR

```

```

130 INT CHR$(145)"_"CHR$(147)CHR
    $(148)
PRINT"_____!_____!____
";: FOR I=1 TO 5: PRINT CHR$(
147);: NEXT: PRINT : PRINT"__
_____!_____!":PRINT: PRIN
T: PRINT: PRINT "!!!!>"; CHR
$(11);
140 VV=5: V=0: T=0: NF=0: GOSUB
    1000
150 PUT SPRITE 1, (255,40),1
160 VPOKE 8194,229: VPOKE 8209,
    165: VPOKE 8210,229: VPOKE
    8211,149: VPOKE 8212,197

```

Control

```

200 D=USR(D)
205 IF VPEEK(6917)>(255-ND) THEN
    VPOKE 6916,INT(RND(1)*85): P=
    P-10: GOSUB 1000
210 IF V=0 THEN 200
220 GOSUB 300
230 IF X<5 OR X>250 THEN 550
240 IF Y<105 THEN 200
250 IF X>85 AND X<140 THEN GOSUB
    350: GOTO 200
260 IF X>160 AND X<210 THEN GOSUB
    450: GOTO 200
270 IF X<30 THEN 400
280 GOSUB 500: GOTO 200

```

Movimiento del acróbata

```

300 XV=3+ND/3: M=50
310 T=T+1: X=VV*T*T/M-XV*T+XI: Y=
    4.9*T*T/M+YI
320 X=INT(X): Y=INT(Y): A$="n"+ST
    R$(N): IF T/3=INT(T/3) THEN P
    LAY"M65086L5510XA$:"N=N-1

```

```

325 IF X>255 THEN X=255
327 IF X<0 THEN X=0
330 PUT SPRITE 0,(X,Y),10
340 RETURN

```

Acróbata en el centro de la red

```

350 IF X>105 AND X<117 THEN P=P+
    200: M$="Estupendo!!_____
    ": PLAY"L8N50L8N45L2N50": GOT
    0 360
355 P=P+100: M$="Buen salto!!_____
    _"
360 VV=INT(RND(1)*10): GOSUB 1000
    : GOSUB 1300: V=0: T=0
370 RETURN

```

Acróbata en los cables de alta tensión

```

400 GOSUB 1200: CLS: PRINT"Acroba
    ta_electrocutado": NH=NH-1: P
    UT SPRITE 0,(100,200): PUT SP
    RITE 1,(250,200)
410 FOR TM=1 TO 3000: NEXT: P=P-
    100: GOTO 100

```

Acróbata en la acera

```

450 M$="Caiste_en_la_cerca_____":
    P=P-50: PLAY"L54M1200S10N74R2
    0N74"
460 VV=INT(RND(1)*10): GOSUB 1000
    : V=0: T=0: NF=NF+1: GOSUB
    1300
470 IF NF>2 THEN 700
480 RETURN

```

Acróbata en el suelo

```

500 M$="AY!-Intentalo_otra_vez_...
    _": P=P-30: NF=NF+1: PLAY"L2N
    7"
510 VV=INT(RND(1)*10): GOSUB 1000
    : GOSUB 1300: V=0: T=0
520 IF NF>2 THEN 700
530 RETURN

```

Acróbata fuera de límites

```

550 GOSUB 1200: CLS: PRINT"El_acr
    obata_cayó_en": PRINT"un_pant
    ano_lejos_del_blanco"
560 PUT SPRITE 0,(200,200): PUT S
    PRITE 1,(200,200): FOR TM=1
    TO 3000: NEXT
570 NH=NH-1: GOTO 100

```

Fin de juego

```

600 CLS: PRINT TAB(6) " FIN_DEL_J
    UEGO": PRINT: PRINT: PRINT
610 PRINT"Tus_puntos_son";P
620 PUT SPRITE 1,(100,200): PUT S
    PRITE 0,(100,200)
630 END

```

Acróbata dado de baja

```

700 V=1: CLS: PRINT"Demasiados_ac
    cidentes": PRINT"Acrobata_dad
    o_de_baja"
710 PUT SPRITE 0,(200,200): PUT S

```

```

PRITE 1,(200,200):NH=NH-1:
FOR TM=1 TO 3000: NEXT
GOTO 100

```

720

Espacio pulsado

```

900 IF V=1 THEN RETURN
910 K1=VPEEK(6916): K2=VPEEK(6917
    ): PUT SPRITE 0,(K2,K1),10:
    V=1: T=0: M$="JERONIMO0000!!_
    _": GOSUB 1300
920 YI=K1: XI=K2: N=50: RETURN

```

Puntuación

```

1000 FOR I=1 TO 22: PRINT: NEXT: P
    RINT"V.del_viento";VV: TAB(13
    );"Puntuacion";P: CHR$(11);:
    F=1
1010 FOR J=6895 TO 6910: VPOKE J,
    32: NEXT
1020 RETURN

```

Acróbata muerto

```

1200 CLS: PLAY"19m1000s14n33n33n33
    n33n33n33n33n33n33n33": FOR I
    15 TO 1 STEP -1: FOR T=1 TO
    100: NEXT: COLOR 15,I,I: NEXT
1250 RETURN

```

Mensaje

```

1300 PRINT CHR$(11);: FOR I=1 TO
    22: PRINT: NEXT: FOR I=1 TO

```

1310 13: PRINT CHR\$(28):: NEXT  
PRINT M\$: CHR\$(11):: RETURN

Datos de sprites

10000 DATA 153,255,60,24,60,36,102,  
195

10010 DATA 0,0,0,112,120,60,30,127,  
255,255,127,1,0,0,0,0,0,0,4  
,14,30,62,254,255,255,254,254  
,240,120,60,28

Datos de sprites pequeños

10020 DATA 170,85,170,85,170,85,170  
,85

10030 DATA 1,2,4,10,17,34,85,138

10040 DATA 128,64,32,80,136,68,170,  
81

10050 DATA 0,0,0,0,0,0,255,36

10060 DATA 56,56,56,56,56,56,56,56

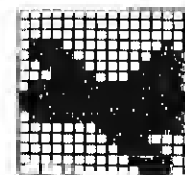
10070 DATA 16,56,56,56,56,56,56,56

10080 DATA 24,60,126,126,126,60,24,  
0

10090 DATA 16,56,84,186,84,186,84,  
16

150	= 975	370	= 143	910	= 9785
160	= 2990	400	= 8338	920	= 1800
200	= 691	410	= 2720	1000	= 6994
205	= 5225	450	= 7268	1010	= 1883
210	= 937	460	= 4454	1020	= 143
220	= 207	470	= 985	1200	= 8089
230	= 1634	480	= 143	1250	= 143
240	= 1043	500	= 7839	1300	= 3965
250	= 2529	510	= 3389	1310	= 1111
260	= 2797	520	= 985	10000	= 1630
270	= 916	530	= 143	10010	= 7656
280	= 1069	550	= 8246	10020	= 1640
300	= 1562	560	= 4089	10030	= 1289
310	= 5721	570	= 1387	10040	= 1575
320	= 8191	600	= 3325	10050	= 1084
325	= 1564	610	= 2222	10060	= 1424
327	= 1074	620	= 2397	10070	= 1420
330	= 832	630	= 129	10080	= 1516
340	= 143	700	= 7380	10090	= 1525
350	= 9578	710	= 5142		
355	= 3975	720	= 489	TOTAL=	321372
360	= 3389	900	= 851		

## Diseño de los sprites



AVION



ACROBATA



## Tabla de claves

5	= 0	40	= 864	105	= 888
6	= 0	50	= 924	110	= 1768
7	= 0	60	= 3607	115	= 21992
10	= 4528	65	= 1572	120	= 15002
15	= 4011	70	= 3295	125	= 6068
20	= 3483	75	= 857	127	= 15547
25	= 4334	90	= 5592	129	= 15158
26	= 4304	95	= 1034	130	= 16583
30	= 2571	100	= 1329	140	= 2283

# Naves secretas 20 de investigación



Tipo de juego:

JUEGO DE TIRO AL BLANCO

Los espías del planeta Circon han robado la fórmula de un componente valioso. El juego consiste en poner agentes de contraespionaje en pequeñas naves que aterricen en lugares especiales del planeta para que la fórmula pueda ser recuperada.

Debes depositar tantas naves como puedas sin rebasar el límite del tiempo, y evitar que la nave nodriza choque con las enemigas.

Se utiliza una subrutina en código máquina para animar la pantalla.



## SUGERENCIAS AL PROGRAMA

Para alterar la velocidad, modificar las líneas:

Nave nodriza            línea 45  
Movimiento de pantalla   línea 160

Puedes cambiar también el límite del tiempo —TL en la línea 160—, para cambiar la duración del juego.

Una modificación más ambiciosa podría consistir en añadir una nave enemiga, con la que se debe evitar chocar; sería un auténtico reto; lo puedes hacer en las líneas 300 a 320.

## Variables del programa

LN   Localización de la nave.  
P   Puntuación.  
TL   Tiempo límite.  
FJ   Final del juego (0, 1, 2).

○			○
○		Inicialización	○
○	10	REM EJECUTA EL SOPORTE	○
○	11	REM EN CODIGO MAQUINA	○
○	15	SCREEN 1,3:KEY OFF: PRINT TAB	○
○		(5) "NAVES_SECRETAS_DE": PRIN	○
○		T TAB(7) "INVESTIGACION": FOR	○
○		I=1 TO 12: PRINT: NEXT	○
○	20	FOR I=1 TO 32: READ Q: A\$=A\$+	○
○		CHR\$(Q) : NEXT: SPRITE\$(0)=A\$	○
○		:A\$=""	○
○	25	FOR I=1 TO 8 : READ Q: A\$=A\$+	○
○		CHR\$(Q) : NEXT :SPRITE\$(1)=A\$	○
○	30	FOR I=1072 TO 1231 : READ Q :	○
○		VPOKE I,Q: NEXT	○
○	35	FOR I= 60350! TO 60401! :READ	○
○		Q : POKE I,Q: NEXT	○
○	40	POKE 60346!,224: POKE 60347!,	○
○		24 : POKE 60348!,255 : POKE	○
○		60349!,26 : DEFUSR2 = 60350!	○
○	45	POKE 59999!,4 : POKE 59996!,	○
○		15 :DEFUSR = 60000!	○

○	50	PUT SPRITE 0, (112,55),11 : PR	○
○		INT "PULSA_UNA_TECLA_PARA_EMP	○
○		EZAR"	○
○	55	IF INKEY\$ = "" THEN Q=USR2(D)	○
○		: FOR I=1 TO 50: NEXT: D=RND	○
○		(1):GOTO 55	○
○			○
○		Preparación pantalla	○
○			○
○	100	COLOR 15,1,1:CLS:VPOKE 8208,	○
○		145:VPOKE 8209,145:VPOKE 8210	○
○		,145: PUT SPRITE 0, (114,80),	○
○		11	○
○	105	VPOKE 6753,137: VPOKE 6768,	○
○		144: VPOKE 6769,145: VPOKE	○
○		6773,134	○
○	110	VPOKE 6784,136: VPOKE 6785,	○
○		134: VPOKE 6786,138: VPOKE	○
○		6787,136: VPOKE 6799,139: VPO	○
○		KE 6800,134: VPOKE 6801,134:	○
○		VPOKE 6802,141: VPOKE 6805,	○
○		134: VPOKE 6813,144: VPOKE	○
○		6814,145	○
○	115	VPOKE 6817,134: VPOKE 6819,	○
○		137: VPOKE 6820,140: VPOKE	○
○		6822,136: VPOKE 6823,148: VPO	○
○		KE 6824,149: VPOKE 6825,148:	○
○		VPOKE 6826,149	○
○	116	VPOKE 6827,138: VPOKE 6829,	○
○		137: VPOKE 6830,140: VPOKE	○
○		6832,142: VPOKE 6833,143: VPO	○
○		KE 6837,134: VPOKE 6838,136	○
○	120	VPOKE 6839,148: VPOKE 6840,	○
○		149: VPOKE 6841,148: VPOKE	○
○		6842,149: VP	○
○	125	VPOKE 6849,134: VPOKE 6850,	○
○		139: VPOKE 6851,134: VPOKE	○
○		6852,134: VPOKE 6853,141: VPO	○
○		KE 6855,150: VPOKE 6856,151:	○
○		VPOKE 6857,150	○
○	126	VPOKE 6858,151: VPOKE 6860,	○
○		139: VPOKE 6861,134: VPOKE	○

```

130 6862,134: VPOKE 6863,141: VPO
    KE 6864,134: VPOKE 6865,134
    VPOKE 6866,144: VPOKE 6867,
    145: VPOKE 6869,134: VPOKE
    6871,150: VPOKE 6872,151: VPO
    KE 6873,150: VPOKE 6874,151:
    VPOKE 6877,144: VPOKE 6878,145
135 VPOKE 6881,134: VPOKE 6883,
    142: VPOKE 6884,143: VPOKE
    6887,150: VPOKE 6888,151: VPO
    KE 6889,150: VPOKE 6890,151:
    VPOKE 6892,139
    VPOKE 6893,134: VPOKE 6894,
    134: VPOKE 6895,141: VPOKE
    6896,134: VPOKE 6897,134
140 VPOKE 6898,142: VPOKE 6899,
    143: VPOKE 6901,134: VPOKE
    6903,150: VPOKE 6904,151: VPO
    KE 6905,150: VPOKE 6906,151:
    VPOKE 6909,146: VPOKE 6910,147
145 FOR I= 6176 TO 6367: IF RND(1
    )<.2 THEN VPOKE I,153
150 NEXT
155 GOSUB 1000: PUT SPRITE 1,(255
    ,100),13
160 TIME = 0: ON INTERVAL= 10 GOS
    UB 500: INTERVAL ON: TL=100
170 STRIG(0) ON: ON STRIG GOSUB
    700
175 SPRITE ON: ON SPRITE GOSUB
    600

```

Control

```

200 D=USR(D): IF VPEEK (6912)>136
    THEN FJ=1: GOTO 900
202 X=VPEEK(6913): IF X<57 THEN P
    OKE 59996!,13 ELSE IF X>184
    THEN POKE 59996!,7 ELSE POKE
    59996!,15
210 GOSUB 300
220 IF LNK>0 THEN GOSUB 400
290 GOTO 200

```

Movimiento nave enemiga

```

300 SPRITE OFF: IF VPEEK(6917)<9
    THEN VPOKE 6916,INT(RND(1)*
    128+24): VPOKE 6917,255
310 SPRITE ON : VPOKE 6917,VPEEK(
    6917)-7
320 RETURN

```

Movimiento nave investigación

```

400 INTERVAL OFF: STRIG(0) OFF: K
    =VPEEK(LN+32): INTERVAL ON
401 IF K=148 OR K=149 THEN VPOKE
    8208,241: VPOKE 8209,241: VPO
    KE 8210,241: P=P+10: PLAY "S1
    M2000L14N50N45":GOSUB 1000: V
    POKE LN,32: LN=0: VPOKE 8208,
    145:TH=1
402 IF TH=1 THEN VPOKE 8209,145:
    VPOKE 8210,145: STRIG(0) ON:
    TH=0: RETURN
410 IF K= 32 THEN LN=LN+32: VPOKE
    LN,152: VPOKE LN-32,32: STRIG
    (0) ON: RETURN

```

Estampido

```

450 VPOKE LN,135: FOR I=1 TO 100:
    NEXT: P=P-10: GOSUB 1000: VPO
    KE LN,32: LN=0: STRIG(0) ON:
    RETURN

```

Movimiento de pantalla

```

500 D=USR2(D): IF TIME/60 >TL THE
    N 900

```

```

505 IF LNK>0 THEN LN=LN-1
510 RETURN

```

Choque

```

600 INTERVAL OFF: STRIG(0) OFF: F
OR I= 1 TO 20: COLOR 15,1,1:P
LAY"164s8m200000n21n12": COLOR
1,15,15: NEXT
610 FJ=2: GOTO 900

```

Dejar nave investigación

```

700 IF LNK>0 THEN RETURN
710 I3=INT((VPEEK(6913)+20)/8): L
N=6144+32*INT(((VPEEK(6912)+
20)/8))+I3: IF LNK<6368 THEN
LN=0
740 RETURN

```

Fin de juego

```

900 COLOR 15,4,7: SCREEN 1: PRINT
"TUS_PUNTOS_SON":P
910 PRINT: PRINT: PRINT: ON (FJ+1
) GOTO 920,930,940
920 PRINT "TUS TIEMPO SE ACABO
": GOTO 990
930 PRINT "COLISIONO LA NAVE NO
DRIZA": GOTO 990
940 PRINT "LA NAVE NODRIZA CHOC
O CON UNA NAVE ENEMIGA
"
990 IF INKEY$="" THEN END ELSE
990

```

Puntuación

```

1000 PRINT "PUNTUACION: ";P;CHR$(11
);
1010 RETURN

```

Datos de sprites

```

10000 DATA 3,4,15,31,63,64,255,213,
255,213,255,63,31,0,0,0,192,
32,240,248,252,2,255,85,255,
85,255,252,248,0,0,0
10010 DATA 160,160,64,160,160,0,0,0

```

Datos de sprites pequeños

```

10100 DATA 255,255,255,255,255,255,
255,255
10102 DATA 68,16,130,40,84,130,16,
68
10104 DATA 192,48,12,3,0,0,0,0
10106 DATA 0,0,0,0,15,31,63,127
10108 DATA 3,12,48,192,0,0,0,0
10110 DATA 1,3,7,15,15,7,3,1
10112 DATA 0,0,0,0,240,248,252,254
10114 DATA 128,192,224,240,240,224,
192,128
10116 DATA 255,127,63,31,31,32,64,
128
10118 DATA 255,254,252,248,248,4,2,
1
10120 DATA 0,0,0,0,3,15,63,255
10122 DATA 0,0,0,0,192,240,252,255
10124 DATA 255,63,15,3,0,0,0,0
10126 DATA 255,252,240,192,0,0,0,0
10128 DATA 0,0,0,0,255,24,31,24
10130 DATA 0,0,0,0,255,24,248,24

```

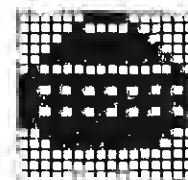
○	10132	DATA 24, 24, 31, 24, 24, 31, 24, 24	○
	10134	DATA 24, 24, 248, 24, 24, 248, 24, 24	
○	10136	DATA 16, 16, 40, 40, 56, 124, 254, 130	○
	10138	DATA 0, 0, 0, 32, 0, 0, 0, 0	
○			○
○		Datos de código máquina	○
○	10200	DATA 6, 31, 42, 186, 235, 43, 35, 205, 27, 235, 120, 254, 31, 202, 215, 235, 4, 43, 205, 44, 235	○
○	10202	DATA 35, 195, 227, 235, 6, 0, 17, 31, 0, 25, 205, 44, 235, 183, 237, 82, 237, 91, 188, 235, 123, 189, 194, 196, 235, 122, 188, 194, 196, 235, 201	○
○			○
○			○

## Tabla de claves

10	= 0	130	= 8073	401	= 9490
11	= 0	135	= 2116	402	= 3378
15	= 6597	136	= 4057	410	= 4114
20	= 3995	140	= 1341	450	= 5193
25	= 3499	145	= 2900	500	= 2295
30	= 2011	150	= 131	505	= 1861
35	= 2217	155	= 1576	510	= 143
40	= 4607	160	= 4187	600	= 5947
45	= 2521	170	= 1684	610	= 1010
50	= 3770	175	= 1014	700	= 1167
55	= 4291	200	= 3266	710	= 7605
100	= 3435	202	= 5778	740	= 143
105	= 2878	210	= 207	900	= 2657
110	= 3089	220	= 1346	910	= 2485
115	= 6750	290	= 593	920	= 4303
116	= 3841	300	= 4717	930	= 4556
120	= 5615	310	= 1556	940	= 8904
125	= 6826	320	= 143	990	= 1529
126	= 5859	400	= 4705	1000	= 1940

1010	= 143	10112	= 1386	10130	= 1274
10000	= 8092	10114	= 1888	10132	= 1398
10010	= 1463	10116	= 1563	10134	= 1523
10100	= 1888	10118	= 1541	10136	= 1568
10102	= 1514	10120	= 1136	10138	= 974
10104	= 1156	10122	= 1378	10200	= 4681
10106	= 1195	10124	= 1147	10202	= 9394
10108	= 1149	10126	= 1410		
10110	= 1035	10128	= 1213	TOTAL=	241020

## Diseño de los sprites



NAVE SECRETA



NAVE ENEMIGA

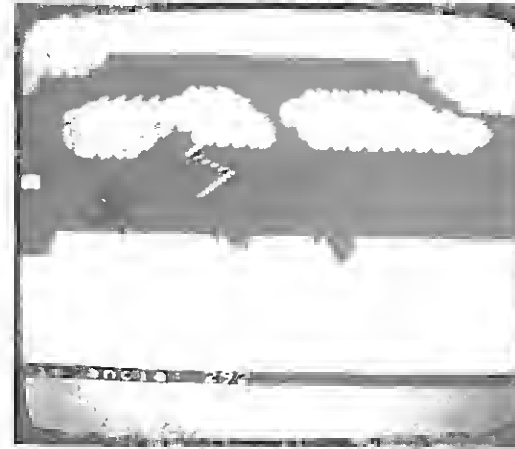


CAPSULA  
(CHARACTER)



# Cross

# 21



Tipo de juego:

JUEGO DE SIMULACION

Este es un juego para corredores entusiastas.

Consigue tu mejor marca personal en los mil metros obstáculos. Para saltar los hoyos pulsa la barra espaciadora, y las flechas para controlar tu velocidad; pero ten cuidado de que no te dé el rayo.

Se usa una subrutina en código máquina para desplazar la pantalla.

## SUGERENCIAS AL PROGRAMA

Si no estas satisfecho con el suelo o con las nubes, modifica la rutina que da movimiento a la pantalla en las líneas 100 a 190.

También se puede cambiar la velocidad usual en la línea 90 (para mover al hombre más lejos) y en la línea 180 (para acelerar el movimiento de pantalla).

## Variables del programa

S Salto.  
DR Distancia recorrida.  
CJ Columna del jugador.  
FJ Final del juego.  
DL Destellos de luz.  
ND Nivel de dificultad.

		Inicialización	
5	REM EJECUTA EL CODIGO		
6	REM MAQUINA SOPORTE		
7	REM DEL PROGRAMA		
10	KEY OFF: COLOR 15,4,7: SCREEN 1,3: FOR I=1 TO 32: READ Q: A\$=A\$+CHR\$(Q): NEXT: SPRITE\$(0)=A\$: A\$=""		
15	FOR I=1 TO 32: READ Q: A\$=A\$+CHR\$(Q): NEXT: SPRITE\$(1)=A\$: A\$=""		
20	FOR I=1 TO 32: READ Q: A\$=A\$+CHR\$(Q): NEXT: SPRITE\$(2)=A\$: A\$=""		
25	FOR I=1 TO 32: READ Q: A\$=A\$+CHR\$(Q): NEXT: SPRITE\$(3)=A\$: A\$=""		
30	FOR I=60350! TO 60401!: READ Q: POKE I,Q: NEXT: DEFUSR2=60350!: POKE 60346!,0: POKE 60347!,24: POKE 60348!,223: POKE 60349!,25		
35	FOR I=1072 TO 1215: READ Q: VPOKE I,Q: NEXT		

50	FOR I=1 TO 11: PRINT : NEXT: PRINT TAB(11)"CROSS"	
55	DEFUSR=60000!: DEFUSR1=60118!: POKE 59999!,1: POKE 59996!,10: POKE 59997!,0: POKE 59998!,3	
60	PUT SPRITE 0,(0,55),1	
65	IF INKEY\$="" THEN IF VPEEK(6914)=0 THEN VPOKE 6914,4 ELSE VPOKE 6914,0 ELSE GOTO 80	
70	D=USR1(D): GOTO 65	
80	PRINT: PRINT: INPUT"Nivel de dificultad_(1-3)":ND	
85	IF ND<1 OR ND>3 THEN 80	
90	POKE 59999!,ND+2	

Preparación pantalla

100	PUT SPRITE 0,(100,200): CLS: FOR I=6609 TO 6879: VPOKE I,134: NEXT	
105	VPOKE 6592,134: VPOKE 6593,134: VPOKE 6594,134: VPOKE 6595,139: VPOKE 6596,130: FOR I=6597 TO 6606: VPOKE I,134: NEXT: VPOKE 6607,139: VPOKE 6609,138	
110	VPOKE 6560,135: VPOKE 6561,135: VPOKE 6562,136: VPOKE 6563,137: FOR I=6565 TO 6574: VPOKE I,135: NEXT: VPOKE 6575,137: FOR I=6578 TO 6586: VPOKE I,135: NEXT: VPOKE 6587,136: VPOKE 6588,137: VPOKE 6589,138	
115	VPOKE 6590,135: VPOKE 6591,135	
120	VPOKE 6152,142: FOR I=6153 TO 6163: VPOKE I,140: NEXT: VPOKE 6164,146: VPOKE 6195,142: FOR I=6186 TO 6192: VPOKE I,140: NEXT: VPOKE 6193,144: VPOKE 6194,144: VPOKE 6195,145	

```

O 125 VPOKE 6209,150:FOR I=6210 TO
O 6215:VPOKE I,149:NEXT:VPOKE
O 6216,147:VPOKE 6218,143:FOR I
O =6219 TO 6223:VPOKE I,140: NE
O XT:VPOKE 6224,146: VPOKE 6233
O ,150:VPOKE 6234,149:VPOKE6235
O ,149:VPOKE 6236,147
O 130 VPOKE6240,141:FORI=6241TO6248
O :VPOKE I,140:NEXT:VPOKE 6249,
O 148:VPOKE 6250,147:VPOKE 6251
O ,143:VPOKE 6252,144:VPOKE6253
O ,144:VPOKE 6254,14:VPOKE 6255
O ,15:VPOKE 6259,150:FOR I=6260
O TO 6263: VPOKE I,151: NE
O 135 VPOKE 6264,151: FOR I=6265 TO
O 6268: VPOKE I,140: NEXT:
O VPOKE 6269,148
O 140 VPOKE 6272,142: FOR I=6273 TO
O 6283: VPOKE I,140: NEXT: VPOK
O E 6284,148: VPOKE 6290,141:
O FOR I=6291 TO 6301: VPOKE I,
O 140: NEXT: VPOKE 6302,148
O 145 VPOKE 6305,142:FOR I=6306 TO
O 6315:VPOKE I,140:NEXT:VPOKE
O 6316,146:VPOKE 6322,141:FOR I
O =6323 TO 6327:VPOKE I,140:NEX
O T:VPOKE 6328,146:VPOKE 6329,
O 143: FOR I=6330 TO 6334: VPOK
O E I,140: NEXT
O 150 VPOKE 6338,143:FOR I=6339 TO
O 6346:VPOKE I,144:NEXT:VPOKE
O 6347,145:VPOKE 6354,142:FOR I
O =6355 TO 6358:VPOKE I,140: NE
O XT:VPOKE 6359,146:VPOKE 6362,
O 143:VPOKE 6363,144:VPOKE 6364
O ,144: VPOKE 6365,144
O 151 VPOKE 6390,145
O 155 VPOKE 6387,143: VPOKE 6388,
O 144: VPOKE 6389,144: VPOKE
O 6390,145
O 170 PUT SPRITE 0,(154,72)
O 180 INTERVAL ON: ON INTERVAL=15-2
O *ND GOSUB 500
O 185 SPRITE ON: ON SPRITE GOSUB
O 800

```

```

O 190 STRIG(0) ON: ON STRIG GOSUB
O 700: TIME=0
O
O Control
O
O 200 D=USR(D): K=INT((VPEEK(6913)+
O 8)/8): IF K-CJ>0 THEN DR=DR+8
O *(K-CJ)
O 205 CJ=K: IF VPEEK(6560+CJ)<>135
O AND S=0 AND CJ<>32 THEN GOSUB
O 600
O 210 IF S<>0 THEN S=S+1: IF S=10-2
O *ND THEN S=0: VPOKE 6914,0
O 220 GOSUB 1000
O 230 IF RND(1)<.07 THEN GOSUB 300
O 240 IF DL<>0 THEN GOSUB 320
O 290 GOTO 200
O
O Rayos
O
O 300 IF DL<>0 THEN RETURN
O 305 PUT SPRITE 3,(INT(RND(1)*255)
O ,50),11: DL=1: PLAY"12m2000s1
O 0n45": RETURN
O 320 DL=DL+1
O 330 IF DL=5 THEN PUT SPRITE 3,(
O 100,200),11: DL=0: RETURN
O 350 RETURN
O
O Movimiento de pantalla
O
O 500 D=USR2(D)
O 510 DR=DR + 1
O 515 IF S<>0 THEN RETURN
O 520 STRIG(0) OFF: I1=VPEEK(6914):
O IF I1=4 THEN I1=0 ELSE I1=4
O 530 VPOKE 6914,I1: STRIG(0) ON:
O RETURN

```

Corredor en un bache

```
600 SPRITE OFF: INTERVAL OFF: STR
    IG(0) OFF: VPOKE 6914,8: VPOKE
    6912,94: PLAY"12m40000s8n4":
    FOR I2=1 TO 1500: NEXT: VPOKE
    6912,72: VPOKE 6914,0
610 VPOKE 6913,(VPEEK(6913)+1)
    MOD 255: CJ=INT((VPEEK(6913)+
    8)/8): IF VPEEK(6560+CJ)<>135
    THEN 610
620 SPRITE ON: INTERVAL ON: STRIG
    (0) ON: RETURN
```

Salto

```
700 IF S<>0 THEN RETURN
710 S=1: VPOKE 6914,8: RETURN
```

Corredor fulminado por un rayo

```
800 INTERVAL OFF: FOR I=1 TO 20:
    PLAY"164m800s14n50": FOR T=1
    TO 50: NEXT: COLOR 15,1,1:
    FOR T=1 TO 50: NEXT: COLOR 15
    ,4,7: NEXT
```

```
810 FJ=1
```

Fin de juego

```
900 CLS: IF FJ=1 THEN PRINT"Te_lha
    s_electrocutado.": PRINT: PRI
    NT"Distancia:":DR;"metros_en"
    : PRINT USING"#####.##":TIME/
```

```
60:: PRINT"_segundos.": GOTO
990
910 PRINT"Se_lacabo_lacarrera.":
    PRINT: PRINT"Tu_tiempo_fue":
    PRINT USING"#####.##": TIME/
    60 : PRINT"_segundos."
990 IF INKEY$<>" " THEN END ELSE
    990
```

Distancia recorrida

```
1000 FOR I=1 TO 23: PRINT: NEXT: P
    RINT"Distancia:":DR;CHR$(11):
1010 IF DR>1000 THEN 900
1020 RETURN
```

Datos de sprites

```
10000 DATA 1,3,3,1,7,11,19,11,7,3,
    195,70,44,24,0,0,128,192,192,
    128,192,196,232,208,192,192,
    96,48,24,16,32,96
10010 DATA 1,3,3,1,3,3,3,3,3,1,1,
    1,2,4,2,128,192,192,128,192,
    192,240,192,192,128,128,128,
    128,128,128,192
10020 DATA 33,19,11,5,3,3,3,3,3,131
    ,255,0,0,0,0,0,132,200,200,
    160,192,192,192,192,192,193,
    255,0,0,0,0,0
10030 DATA 6,12,24,48,12,3,0,0,0,0,
    0,0,0,1,3,6,0,0,0,0,0,192,
    48,12,24,48,96,192,128,0,0
```

Datos de código máquina

```
10100 DATA 6,31,42,186,235,43,35,
    205,27,235,120,254,31,202,215
```



, 235, 4, 43, 205, 44, 235, 35, 195,  
227, 235, 6, 0, 17, 31, 0, 25, 205, 44  
, 235, 183, 237, 82, 237, 91, 188,  
235, 123, 189, 194, 196, 235, 122,  
188, 194, 196, 235, 201

### Datos de sprites pequeños

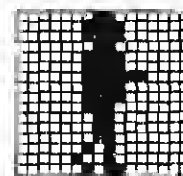
10200 DATA 255, 255, 255, 255, 255, 255,  
255, 255  
10202 DATA 0, 170, 255, 255, 255, 255,  
255, 255  
10204 DATA 0, 192, 225, 227, 243, 247,  
247, 255  
10206 DATA 0, 0, 128, 192, 192, 224, 224,  
240  
10208 DATA 0, 2, 3, 3, 7, 7, 7, 15  
10210 DATA 240, 240, 248, 248, 252, 252,  
254, 255  
10212 DATA 170, 85, 170, 85, 170, 85, 170  
, 85  
10214 DATA 1, 5, 10, 13, 21, 26, 13, 21  
10216 DATA 10, 21, 10, 13, 6, 10, 5, 2  
10218 DATA 170, 85, 106, 53, 10, 0, 0, 0  
10220 DATA 170, 85, 170, 205, 134, 0, 0, 0  
10222 DATA 85, 170, 84, 172, 80, 0, 0, 0  
10224 DATA 85, 170, 86, 168, 84, 168, 80,  
160  
10226 DATA 0, 0, 0, 160, 80, 176, 84, 170  
10228 DATA 128, 80, 160, 64, 168, 80, 170  
, 84  
10230 DATA 0, 0, 140, 198, 85, 170, 85,  
170  
10232 DATA 0, 0, 0, 2, 1, 5, 10, 13  
10234 DATA 1, 5, 10, 85, 170, 85, 170, 85

### Tabla de claves

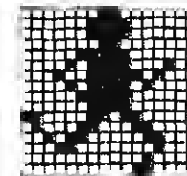
5	= 0	10	= 5594	25	= 3481
6	= 0	15	= 3996	30	= 7334
7	= 0	20	= 3997	35	= 1995

50	= 2213	220	= 397	1020	= 143
55	= 5636	230	= 1589	10000	= 7802
60	= 754	240	= 1258	10010	= 7586
65	= 3961	290	= 593	10020	= 7182
70	= 1255	300	= 1159	10030	= 5457
80	= 3303	305	= 4449	10100	= 21635
85	= 1566	320	= 801	10200	= 1888
90	= 1135	330	= 2603	10202	= 1778
100	= 3295	350	= 143	10204	= 1780
105	= 8634	500	= 716	10206	= 1625
110	= 13438	510	= 815	10208	= 965
115	= 1362	515	= 1089	10210	= 1081
120	= 8848	520	= 3762	10212	= 1640
125	= 3696	530	= 1423	10214	= 1265
130	= 4661	600	= 8074	10216	= 1202
135	= 3257	610	= 6968	10218	= 1331
140	= 7762	620	= 2515	10220	= 1470
145	= 8856	700	= 1089	10222	= 1344
150	= 512	710	= 939	10224	= 1657
155	= 560	800	= 7565	10226	= 1393
170	= 836	810	= 406	10228	= 1651
180	= 3782	900	= 12082	10230	= 1518
185	= 926	910	= 7331	10232	= 1029
190	= 2240	990	= 1777	10234	= 1399
200	= 6086	1000	= 3516		
205	= 4677	1010	= 1190	TOTAL=	284505
210	= 3987				

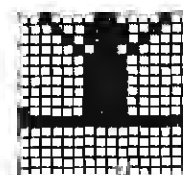
### Diseño de los sprites



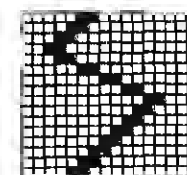
CORREDOR: POSICION 2



CORREDOR: POSICION 1



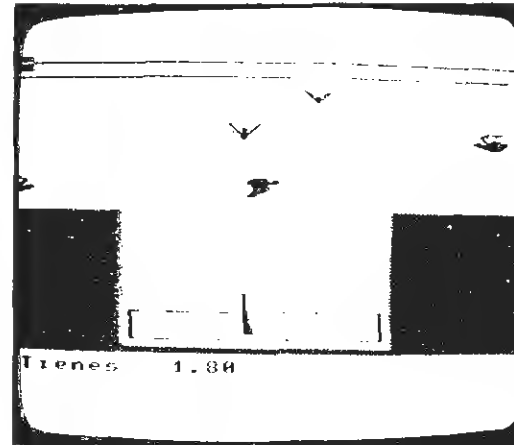
CORREDOR: CAIDA



RAYO



# Caseta de tiro 22



Tipo de juego:

TIRO AL BLANCO

Prueba tu puntería. ¡Cinco tiros por sólo cien pesetas!

Puntuación:

Patos cercanos	20 pesetas.
Patos centrales	50 pesetas.
Pájaro loco	100 pesetas.

## SUGERENCIAS AL PROGRAMA

Se puede hacer más difícil alcanzar a los pájaros modificando las líneas 710-750 y su velocidad se controla en la línea 350 (pájaros centrales) y la línea 410 (cercanos).

Intenta añadir pájaros con distintos valores para dar más variedad.

## Variables del programa

DR Dinero restante.  
 NT Número de tiros (0-5).  
 PH Pájaro herido (-1 si no está herido).  
 C1, C2 Espacio de tiempo entre los pájaros cercanos y centrales.  
 D ¿Te queda dinero?

### Inicialización

```

10 REM EJECUTA EL PROGRAMA
11 REM SOPORTE EN CODIGO
12 REM MAQUINA
15 SCREEN 1,2: COLOR 1,15,15: KE
    Y OFF: PRINT TAB(7)"CASETA_DE
    _TIRO"
20 FOR I=1 TO 32: READ Q: A$=A$+
    CHR$(Q): NEXT: SPRITE$(0)=A$
    : A$=""
25 FOR I=1 TO 32: READ Q: A$=A$+
    CHR$(Q): NEXT: SPRITE$(1)=A$
    : A$=""
30 FOR I=1 TO 32: READ Q: A$=A$+
    CHR$(Q): NEXT: SPRITE$(2)=A$
    : A$=""
35 FOR I=1 TO 32: READ Q: A$=A$+
    CHR$(Q): NEXT: SPRITE$(3)=A$
    : A$=""
40 FOR I=1 TO 32: READ Q: A$=A$+
    CHR$(Q): NEXT: SPRITE$(4)=A$
    : A$=""
45 FOR I=1 TO 32: READ Q: A$=A$+
    CHR$(Q): NEXT: SPRITE$(5)=A$
    : A$=""
    
```

```

50 FOR I=1 TO 32: READ Q: A$=A$+
    CHR$(Q): NEXT: SPRITE$(6)=A$
    : A$=""
55 FOR I=1072 TO 1119: READ Q: V
    POKE I,Q: NEXT
60 DEFUSR=60118!: POKE 59997!,2:
    POKE 59998!,1: POKE 59999!,5:
    PUT SPRITE 2,(255,80),4
65 D=USR(D): IF INKEY$("<>") THEN
    80
70 D=RND(1): IF VPEEK(6922)=8 TH
    EN VPOKE 6922,12 ELSE VPOKE
    6922,8
75 FOR I=1 TO 120: NEXT: GOTO 65
80 PH=-1
    
```

### Preparación pantalla

```

100 CLS: VPOKE 6922,8: PUT SPRITE
    2,(255,36),12: PUT SPRITE 3,(
    140,36),12: PUT SPRITE 0,(100
    ,16),8: PUT SPRITE 4,(0,64),4
    : PUT SPRITE 5,(115,64),4
110 FOR I=6145 TO 6206: VPOKE I,
    23: NEXT: FOR I=6729 TO 6742:
    VPOKE I,23: NEXT: FOR I=6793
    TO 6806: VPOKE I,23: NEXT
115 FOR I=6816 TO 6847: VPOKE I,
    139: NEXT: FOR I=6496 TO 6784
    STEP 32: FOR J=0 TO 7: VPOKE
    I+J,134: VPOKE I+J+24,134: NE
    XT: NEXT
120 VPOKE 6144,24: VPOKE 6728,24:
    VPOKE 6176,26: VPOKE 6792,26:
    VPOKE 6175,25: VPOKE 6743,25:
    VPOKE 6207,27: VPOKE 6807,27
125 VPOKE 6760,22: VPOKE 6775,22:
    VPOKE 6703,135: VPOKE 6735,
    136: VPOKE 6767,137
130 STRIG(0) ON: ON STRIG GOSUB
    700
140 GOSUB 800
    
```

Control

```
200 GOSUB 300: GOSUB 350: GOSUB
    400
210 IF PH>=0 THEN GOSUB 450
220 IF MD=1 THEN GOSUB 800: MD=0
290 GOTO 200
```

Movimiento del pájaro loco

```
300 POKE 59999!,7: POKE 59997!,0:
    IF RND(1)<.5 THEN POKE 59998!
    ,1 ELSE POKE 59998!,3
320 IF VPEEK(6913)>194 THEN POKE
    59998!,1 ELSE IF VPEEK(6913)<
    65 THEN POKE 59998!,3
330 D=USR(D)
335 IF RND(1)<.5 THEN VPOKE 6914,
    4 ELSE VPOKE 6914,0
340 RETURN
```

Movimiento de pájaros de perfil

```
350 POKE 59999!,13: POKE 59997!,2
    : K1=VPEEK(6922): VPOKE 6922,
    VPEEK(6926): VPOKE 6926,K1
360 POKE 59998!,1: D=USR(D): POKE
    59997!,3: D=USR(D)
370 RETURN
```

Movimiento de pájaros de frente

```
400 C2=(C2+1) MOD 2: POKE 59997!,
    4: IF C2=0 THEN K1=VPEEK(6930
    ): VPOKE 6930,VPEEK(6934): VP
    OKE 6934,K1
```

```
410 POKE 59999!,4: POKE 59998!,3:
    D=USR(D): POKE 59997!,5: D=US
    R(D)
430 RETURN
```

Más pájaros

```
450 ON (PH+1) GOTO 465,465,470,
    475,480,485
465 PUT SPRITE 0,(100,16): GOTO
    490
470 VPOKE 6922,8: VPOKE 6926,12
472 GOTO 490
475 VPOKE 6922,8: VPOKE 6926,12
477 GOTO 490
480 VPOKE 6930,16: VPOKE 6934,20
483 GOTO 490
485 VPOKE 6930,16: VPOKE 6934,20
490 PH=-1: RETURN
```

Pájaro derribado

```
500 PLAY"L50M1200S10N74R30N74R30N
    74R30N74R30N74R30N74R30N74":
    ON (PH+1) GOTO 505,508,510,
    515,520,525
505 DR=DR+1: VPOKE 6914,24: FOR I
    2=1 TO 500: NEXT
508 VPOKE 6914,0: GOTO 540
510 DR=DR+.5: VPOKE 6922,24: FOR
    I2=1 TO 500: NEXT
512 VPOKE 6922,8: GOTO 540
515 DR=DR+.5: VPOKE 6926,24: FOR
    I2=1 TO 500: NEXT
517 VPOKE 6926,12: GOTO 540
520 DR=DR+.2: VPOKE 6930,24: FOR
    I2=1 TO 500: NEXT
522 VPOKE 6930,16: GOTO 540
525 VPOKE 6934,24: DR=DR+.2: FOR
    I2=1 TO 500: NEXT
527 VPOKE 6934,20
540 MD=1: STRIG(0) ON: RETURN
```

Pájaro fallado

```
600  STRIG(0) ON: VPOKE 6191,138:
      FOR I1=1 TO 100: NEXT: VPOKE
      6191,23
620  MD=1: RETURN
```

Disparo

```
700  STRIG(0) OFF: PLAY"L64M60S8N2
      7": NT=NT+1
710  IF ABS(VPEEK(6929)-114)<3 THE
      N PH=4: GOTO 500
720  IF ABS(VPEEK(6933)-114)<3 THE
      N PH=5: GOTO 500
730  IF ABS(VPEEK(6921)-114)<3 THE
      N PH=2: GOTO 500
740  IF ABS(VPEEK(6925)-114)<3 THE
      N PH=3: GOTO 500
750  IF ABS(VPEEK(6913)-114)<3 THE
      N PH=0: GOTO 500
760  GOTO 600
```

Dinero ganado/perdido

```
800  IF DR>=0 THEN A$="Tienes:" EL
      SE A$="Debes:"
805  IF DR<0 THEN I3=INT(-DR): I4=
      -(DR+I3) ELSE I3=INT(DR): I4=
      DR-I3
807  FOR I=1 TO 22: PRINT: NEXT: P
      RINT A$: PRINT USING"###";I3
      ;: PRINT USING".##";I4: PRINT
      CHR$(11);
810  IF NT>4 THEN 850 ELSE RETURN
```

¿Más disparos?

```
850  PLAY"L10S14M2000N30N40N20N27"
      : STRIG(0) OFF: FOR I=1 TO 23
```

```
855  : PRINT : NEXT: PRINT "5,tiro
      s_mas?(S/N)";CHR$(11);
      X$=INKEY$: IF X$="" THEN 855
      ELSE IF X$<>"S" AND X$<>"s" A
      ND X$<>"N" AND X$<>"n" THEN
      55
860  IF X$="N" OR X$="n" THEN 900
865  DR=DR-1: NT=0: GOSUB 800
870  FOR I=6882 TO 6910: VPOKE I,
      32: NEXT: STRIG(0) ON: RETURN
```

Fin de juego

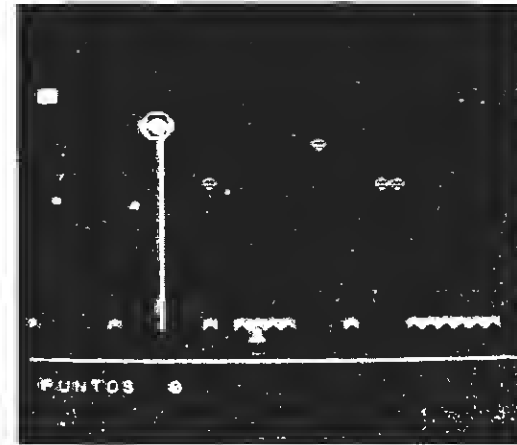
```
900  SCREEN 1: IF DR<0 THEN PRINT"
      Debes: ";: PRINT USING"###.##"
      ;:DR: GOTO 990
920  PRINT "Has_ganado: ";: PRINT U
      SING"###.##";DR: GOTO 990
990  IF INKEY$<>" " THEN END ELSE
```

Datos de sprites

```
10000 DATA 128,64,96,56,28,13,7,3,1
      ,2,0,0,0,0,0,0,1,2,6,28,56,
      176,224,192,128,64,0,0,0,0,0,
      0
10010 DATA 0,0,0,0,0,1,7,15,25,50,
      32,32,32,0,0,0,0,0,0,0,128,
      224,240,152,76,4,4,4,0,0,0
10020 DATA 0,0,0,0,3,14,24,12,6,115
      ,255,63,7,0,0,0,0,0,0,252,
      32,64,32,248,255,255,254,252,
      248,0,0
10030 DATA 0,0,0,0,1,115,255,63,3,7
      ,7,3,0,0,0,0,0,0,0,0,248,254,
      255,255,252,248,248,248,252,
      62,15,0
10040 DATA 0,0,0,0,63,4,2,4,63,255,
      255,127,63,31,0,0,0,0,0,0,192
      ,112,24,48,224,206,255,252,
      224,0,0,0
```



# Batalla laser 23



Tipo de juego:

JUEGO DE DISPARO

Este es uno de los juegos más rápidos y que con deslumbrantes efectos laser y super-sonidos conmoverá a los más entusiastas.

Naves guardianas tratan de arrasar tu planeta mediante la destrucción de uno de los tres acumuladores de energía de tu cañón laser.

La nave nodriza provee de energía a las demás, y, si la destruyes, te suma más puntos.

Para crear los efectos laser se usa una subrutina en código máquina.

Emplea la barra espaciadora para disparar y las teclas de cursor para mover el cañón a derecha e izquierda.

## SUGERENCIAS AL PROGRAMA

Para dar velocidad a la nave nodriza, modifica el POKE de la línea 255 a 59999.

Se pueden añadir más o menos naves guardianas, para lo cual debes corregir las rutinas "Posición de las naves guardianas", "Fuego de las naves" y "Vuelo de las naves".

### Variables del programa

FA Fase (02)  
 AN(N, I) Atributos de las naves:  
     N = Número de nave.  
     I = 1 (Columna).  
     I = 2 (Fila).  
     I = 3 (Señal de parada).  
 CC Columna del cañón del jugador.  
 SN Señal nave herida.  
 PP Punto de partida del láser.  
 NN Número de la nave que está disparando.  
 NC Número de cañones que quedan.

		Inicialización	
5	KEY OFF :SCREEN 1,2 : COLOR 15,1,1		
10	REM EJECUTA EL PROGRAMA		
11	REM SOPORTE EN CODIGO		
12	REM MAQUINA		
13	REM VER APENDICES		
15	FOR I=1 TO 8 : READ Q : A\$=A\$+CHR\$(Q) : NEXT : SPRITE\$(0) = A\$ : A\$=""		
20	FOR I=1 TO 32 : READ Q : A\$=A\$+CHR\$(Q) : NEXT : SPRITE\$(1) =A\$		
25	DEFUSR0=60000!:DEFUSR1=60118!:POKE 59996!,10: POKE 59997!,1 :POKE 59998!,3: POKE 59999!,4:DEFUSR2=60220!:FOR I=60220! TO 60248!: READ Q : POKE I,Q: NEXT		
27	FOR I= 1088 TO 1344 STEP 64:		

		FOR J=0 TO 7 : READ Q : VPOKE I+J,Q : NEXT : NEXT	
30	PUT SPRITE 1,(100,45),7:PUT SPRITE 0,(127,100),11:NC=3: ON STRIG GOSUB 150		
35	CLS : PRINT ".....BATALLA LASER": FOR I =1 TO 18 : PRINT : NEXT : PRINT ".....DA_A_CU ALQUIER_TECLA_FARA_EMPEZAR"		
40	IF INKEY\$="" THEN D=USR1(D): FOR I=1 TO 100 : NEXT :LAY "L9M1000S14N33": GOTO 35		
		Preparación muro protector	
50	CLS : PUT SPRITE 1,(100,200)		
52	VPOKE 8209,225: VPOKE 8210,49 : VPOKE 8211,177:VPOKE 8212,97 : VPOKE 8213,241		
55	FOR I=6720 TO 6751 :VPOKE I,136 : NEXT : FOR I= 6816 TO 6847 : VPOKE I,23 : NEXT :VPOKE 6787,160: VPOKE 6799,160 : VPOKE 6812,160		
57	CC=15 :GOSUB 1000 : PUT SPRITE 0,(120,150),15: STRIG(0) ON		
		Control	
60	POKE 59999!,8 : D=USR0(D):CC=INT(VPEEK(6913)/8)		
65	IF FA=2 THEN GOSUB 400 ELSE IF FA=0 THEN GOSUB 220 : GOTO 70		
67	GOSUB 250		
70	IF RND(1)<.01 AND FA=1 THEN FA=2		
75	IF FA=0 THEN 80		
77	IF RND(1)<.8 OR FA=2 THEN 80		
78	GOSUB 190		
80	IF VPEEK(6912)=200 THEN 500		
90	GOTO 60		



Disparos del jugador

```

150 D=RND(1) : STRIG(0) OFF :PLAY
    "19m1000s14n33"
155 SN=0:I2=0:FOR I1=1 TO 4 : IF
    AN(I1,1)=CC THEN I2=I1
156 NEXT : POKE 59991!,152
157 IF I2<>0 THEN PF=6144+32*AN(
    I2,2)+CC :SN=2: GOTO 165
160 IM=VPEEK (6917) : IF=CC*8
162 IF IF<IM+2 OR IF-IM>6 OR VFEE
    K (6916)=200 THEN PF=6144+CC:
    GOTO 165
163 SN=1:PF=6208+CC
165 POKE 59992!,PF MOD 256 : POKE
    59993!,INT(PF/256): I3=6720 +
    CC: POKE 59994!,I3 MOD 256: P
    OKE 59995!,INT(I3/256):D=USR2
    (D)
170 POKE 59991!,32:FOR I3=1 TO 20
    : NEXT : D=USR2(D)
175 IF SN=1 THEN GOSUB 300
177 IF SN=2 THEN GOSUB 350
180 STRIG(0) ON : RETURN

```

Disparo nave guardiana

```

190 IF AN(1,1)=0 AND AN(2,1)=0
    AND AN(3,1)=0 AND AN(4,1)=0
    THEN RETURN ELSE STRIG(0) OFF
191 NN=NN+1: IF NN=5 THEN NN=1
192 IF AN(NN,1)=0 THEN 191
193 FL=0 : K1=0 :POKE 59991!,152
    : I=6176+32*AN(NN,2)+AN(NN,1)
    : POKE 59992!,I MOD 256 : POK
    E 59993!,INT(I/256)
195 IF VPEEK(6720+AN(NN,1))=136 T
    HEN J=6752+AN(NN,1) : GOTO
    200
196 J=6816 + AN(NN,1) : IF AN(NN,
    1)=CC THEN K1=1

```

```

197 K=AN(NN,1) : IF K=3 OR K=15
    OR K=28 THEN FL=1
200 POKE 59994!,J MOD 256: POKE
    59995!,INT(J/256): D=USR2(D)
    : IF FL=1 THEN FOR J=1 TO 10
    : COLOR 1,15,15 : FOR T=1 TO
    30 :NEXT : PLAY "139m59000s8n
    2":COLOR 15,1,1 : FOR T= 1 TO
    30 :NEXT : NEXT : GOTO 5
202 IF K1=1 THEN PUT SPRITE 0,(
    100,200) : FOR K=1 TO 7 : PLA
    Y "164m1000s14n20n21n20n21":
    NEXT
205 PLAY "119m380s10n50":POKE
    59991!,32:D=USR2(D)
210 STRIG(0) ON : RETURN

```

Posición de la nave guardiana

```

220 IF AN(1,1)<>0 THEN 225
221 I=INT(RND(1)*31+1): IF I=AN(2
    ,1) OR I=AN(3,1) OR I=AN(4,1)
    THEN 221 ELSE AN(1,1)=I : AN(
    1,2)=0 : VPOKE 6144+I,144: RE
    TURN
225 IF AN(2,1)<>0 THEN 230
226 I=INT(RND(1)*31+1) : IF I=AN(
    1,1) OR I=AN(3,1) OR I=AN(4,1
    ) THEN 226 ELSE AN(2,1)=I :
    AN(2,2)=0 : VPOKE 6144+I,144
    : RETURN
230 IF AN(3,1)<>0 THEN 235
231 I=INT(RND(1)*31+1) : IF I=AN(
    1,1) OR I=AN(2,1) OR I=AN(4,1
    ) THEN 231 ELSE AN(3,1)=I :
    AN(3,2)=0 : VPOKE 6144+I,144:
    RETURN
235 IF AN(4,1)<>0 THEN 240
236 I=INT(RND(1)*31+1) : IF I=AN(
    1,1) OR I=AN(2,1) OR I=AN(3,1
    ) THEN 236 ELSE AN(4,1)=I : A
    N(4,2)=0 : VPOKE 6144+I,144:
    RETURN

```

```

240 IF AN(1,3)=1 AND AN(2,3)=1
    AND AN(3,3)=1 AND AN(4,3)=1
    THEN FA=1: RETURN
242 J=INT(RND(1)*4+1) : IF AN(J,3)
    =1 THEN 242
243 VPOKE AN(J,1)+32*AN(J,2)+6144
    ,32 : AN(J,2)=AN(J,2)+1
244 IF AN(J,1)<> 0 THEN VPOKE AN(
    J,1)+32*AN(J,2)+6144,144
246 IF RND(1)<.1 OR AN(J,2)>=7
    THEN AN(J,3)=1
248 RETURN

```

Movimiento nave nodriza

```

250 IF VPEEK(6916)<>200 THEN 255
253 MC=2 : PUT SPRITE 1,(0,15):
    RETURN
255 POKE 59997!,1 : POKE 59998!,3
    :POKE 59999!,6: D=USR1(D):IF
    VPEEK (6917)>251 THEN PUT SPR
    ITE 1,(200,200):RETURN
260 MC=INT((VPEEK(6917)+8)/8): IF
    RND(1)<.85 THEN RETURN
265 IF MC<>AN(1,1) AND MC<>AN(2,1
    ) AND MC<>AN(3,1) AND MC<>AN(
    4,1) THEN GOSUB 280
270 RETURN

```

Disparo nave nodriza

```

280 PLAY "I24m160s8n67"
282 STRIG(0) OFF : FL=0 : K1 = 0
    : POKE 59991!,152 : I=6240+MC
    : POKE 59992!, I MOD 256 : PO
    KE 59993!,INT(I/256)
284 IF VPEEK (MC +6720) =136 THEN
    J=6752+MC : GOTO 290
286 J=6816 +MC : IF MC = 3 OR MC
    = 15 OR MC = 28 THEN FL = 1
287 IF MC = 00 THEN K1= 1
290 POKE 59994!, J MOD 256 : POKE

```

```

59995!,INT(J/256): D = USR2
(D)
291 IF FL=1 THEN FOR J=1 TO 10 :
    COLOR 1,15,15 : FOR T =1 TO
    30 : NEXT: PLAY "I39m59000s8n
    2": COLOR 15,1,1 : FOR T =1
    TO 30 : NEXT : NEXT : GOTO
    500
292 PLAY "I24m160s8n67"
293 IF K1=1 THEN PUT SPRITE 0,(
    100,200) : FOR J= 1 TO 7 : P
    LAY "I64m1000s14n20n21n20n21"
    : NEXT
294 POKE 59991!,32 : D=USR2(D)
299 STRIG(0) ON : RETURN

```

Nave nodriza derribada

```

300 SC = SC+50 : GOSUB 1000
310 PLAY "I64m60000s8n20n21n24n28
    n40n45n43n29n20n1513n10"
320 FOR I4=1 TO 60 : VPOKE 14368
    + INT(RND(1)*30), INT(RND(1)*
    255) : NEXT : SPRITE$(1)=A$
330 PUT SPRITE 1,(200,200)
340 RETURN

```

Nave guardiana derribada

```

350 VPOKE 6144+AN(I2,1)+32* AN(I2
    ,2),168: SC = SC + 10 : GOSUB
    1000 : FOR T = 1 TO 30 : NEXT
355 VPOKE 6144+AN(I2,1)+32*AN(I2,
    2),32 : AN(I2,1)=0
360 RETURN

```

Posición de naves guardianas  
(fase anterior)

```

400 IF AN(1,1)<> 0 OR AN(2,1)<>0
    OR AN(3,1)<> 0 OR AN(4,1)<>0
    THEN STRIG(0) OFF : GOTO 405

```

```

402 FOR J= 1 TO 4 : AN(J,3)=0 :
NEXT : FA =0 : RETURN
405 J= INT(RND(1)*4+1) : IF AN (J
,1)=0 THEN 405 ELSE STRIG(0)
ON
407 VPOKE 6144 +AN(J,1)+32 * AN (
J,2),32 : IF AN(J,2) > 0 THEN
AN(J,2)= AN(J,2)-1
410 VPOKE 6144 + AN(J,1) + 32 *
AN(J,2),144 : IF AN(J,2) = 0
THEN VPOKE 6144 + AN(J,1) ,
32: AN( J,1)= 0
415 RETURN

```

Nuevo juego

```

500 IF VPEEK(6912)<>200 THEN 560
502 STRIG(0) OFF : FOR T=1 TO
1500 : NEXT
510 CLS : PUT SPRITE 0,(100,200)
: PUT SPRITE 1,(100,200): PR
INT "____BATALLA_LASER": PR
INT : PRINT : PRINT
515 NC=NC-1 : IF NC=0 THEN 600
520 PRINT "____HOMBRES_MUERTOS:_
_"; NC
530 FOR J=1 TO 2500: NEXT
535 FOR J=1 TO 4 : AN(J,1) = 0 :
NEXT : FA = 0
550 GOTO 50
560 STRIG(0) OFF: FOR T=1 TO 1500
: NEXT
565 CLS : PUT SPRITE 0,(100,200)
: PUT SPRITE 1,(100,200) : PR
INT "____BATALLA_LASER":
PRINT : PRINT : PRINT
570 PRINT "!!!PLANTA_DESTRUIDA!!!
":PRINT:PRINT:GOTO 515

```

Fin de juego

```

600 PRINT : PRINT : PRINT

```

```

610 PRINT "____TUS_PUNTOS";SC
620 IF INKEY$="" THEN END ELSE
620

```

Puntuación

```

1000 FOR I=1 TO 22: PRINT : NEXT :
PRINT : PRINT "PUNTOS:";SC;
CHR$(11);
1010 RETURN

```

Datos de sprites

```

10000 DATA 24,60,24,60,24,52,122,
255
10010 DATA 15,16,32,32,99,198,203,
254,255,207,199,99,33,48,16,
15,240,8,4,4,198,99,211,127,
255,243,227,198,132,12,8,240

```

Datos de código máquina

```

10100 DATA 42,98,234,58,87,234,79,
205,44,235,17,32,0,25,237,91,
90,234,124,186,194,67,235,125
,187,194,67,235,201

```

Datos de sprites pequeños

```

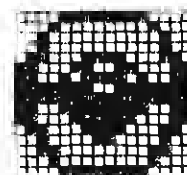
10200 DATA 90,165,90,165,90,165,66,
129
10202 DATA 24,126,213,171,255,66,60
,24
10204 DATA 24,24,24,24,24,24,24,24
10206 DATA 24,36,90,165,219,165,219
,255
10208 DATA 2,144,4,17,64,4,161,8

```

## Tabla de claves

5	= 1217	193	= 7674	310	= 3793
10	= 0	195	= 4334	320	= 5613
11	= 0	196	= 3161	330	= 1012
12	= 0	197	= 3328	340	= 143
13	= 0	200	= 13830	350	= 5153
15	= 4011	202	= 4889	355	= 2961
20	= 3483	205	= 2819	360	= 143
25	= 10602	210	= 883	400	= 6488
27	= 3640	220	= 1467	402	= 2349
30	= 3648	221	= 9505	405	= 4238
35	= 7080	225	= 1466	407	= 4838
40	= 4737	226	= 9491	410	= 5648
50	= 1145	230	= 1465	415	= 143
52	= 2806	231	= 9499	500	= 1724
55	= 4952	235	= 1480	502	= 2018
57	= 2883	236	= 9505	510	= 5059
60	= 3457	240	= 4593	515	= 1765
65	= 3159	242	= 3368	520	= 2078
67	= 410	243	= 3515	530	= 1123
70	= 2389	244	= 3599	535	= 2134
75	= 870	246	= 3286	550	= 443
77	= 2197	248	= 143	560	= 2018
78	= 350	250	= 1914	565	= 5059
80	= 1663	253	= 1329	570	= 2807
90	= 449	255	= 6489	600	= 561
150	= 3068	260	= 3902	610	= 1502
155	= 3701	265	= 5842	620	= 1416
156	= 935	270	= 143	1000	= 3246
157	= 4217	280	= 1195	1010	= 143
160	= 1863	282	= 6939	10000	= 1529
162	= 5427	284	= 3313	10010	= 9019
163	= 1438	286	= 3751	10100	= 8191
165	= 8740	287	= 1287	10200	= 1634
170	= 2535	290	= 3572	10202	= 1649
175	= 998	291	= 7810	10204	= 1404
177	= 1048	292	= 1195	10206	= 1689
180	= 883	293	= 4888	10208	= 1275
190	= 5073	294	= 1399		
191	= 2115	299	= 883		
192	= 1309	300	= 1335		
				TOTAL=	370992

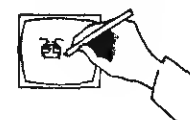
## Diseño de los sprites



NAVE NODRIZA



LASER





## SUGERENCIAS AL PROGRAMA

Hay dos tácticas principales en este juego: ir derecho por los diamantes o pasar primero por una esquina a recoger algunos rubíes. Puedes aumentar las opciones técnicas reorganizando la pantalla y añadiendo algunas gamas diferentes. No es un trabajo difícil, ya que la disposición de la pantalla se controla únicamente en la rutina que va de la línea 100-195.

### Variables del programa

PJ(1) Posición jugador 1.  
P(1) Puntuación jugador 1.  
P\$(1) Nombre del jugador.  
M\$(1) Mensaje a los jugadores.  
JA Jugador en acción.  
CJ Caracteres bajo el jugador.

		Inicialización	
5	COLOR 15,4,7: SCREEN 1,2: KEY OFF		
7	VPOKE 8210,180: VPOKE 8211,36: VPOKE 8212,132: VPOKE 8213,20		
10	FOR I=1 TO 8: READ Q: A\$=A\$+CHR\$(Q): NEXT: SPRITE\$(0)=A\$: SPRITE\$(1)=A\$		
15	FOR I=1072 TO 1079: READ Q: VPOKE I,Q: NEXT: FOR I=1152 TO 1344 STEP 64: FOR J=0 TO 7: READ Q: VPOKE I+J,Q: NEXT: NEXT: FOR I=1112 TO 1143: READ Q: VPOKE I,Q: NEXT		
20	PRINT"_____BUSCA_DEL_TESORO"		
25	PRINT" Gema Valor (millones)";: PRINT" -----"		
30	PRINT: PRINT: PRINT"Opalos" TAB(11)CHR\$(144)TAB(19)"250"		
35	PRINT: PRINT"Esmeraldas" TAB(		

		11)CHR\$(152)TAB(19)"500"	
40	PRINT: PRINT"Rubies" TAB(11)CHR\$(160)TAB(19)"750"		
45	PRINT: PRINT"Diamantes"TAB(11)CHR\$(139)CHR\$(140)TAB(18)"1000":PRINT TAB(11)CHR\$(141)CHR\$(142)		
47	PRINT: PRINT"Serpiente" TAB(11)CHR\$(168)TAB(18)"-100"		
50	PRINT: PRINT: P\$(1)="RICARDO_": INPUT"Jugador_1": P\$(1): PRINT: P\$(2)="JULIO_": INPUT"Jugador_2": P\$(2)		
55	IF LEN(P\$(2))>9 OR LEN(P\$(1))>9 THEN PRINT"MUY_LARGO": GOT 50 ELSE PRINT: PRINT"VALE:_" P\$(1) "_Y_" P\$(2)		
57	FOR I=1 TO 20: PRINT: NEXT: PRINT"_____PULSA_UNA_TECLA"		
60	IF INKEY\$="" THEN D=RND(1): GOT 60		
		Preparación pantalla	
100	CLS: VPOKE 6511,139: VPOKE 6512,140: VPOKE 6543,141: VPOKE 6544,142		
105	FOR I=6241 TO 6244: VPOKE I,134:NEXT: FOR I=6267 TO 6270: VPOKE I,134: NEXT: VPOKE 6273,134: FOR I=6276 TO 6287: VPOKE I,134: NEXT: FOR I=6289 TO 6299: VPOKE I,134: NEXT: VPOKE 6302,134		
110	VPOKE 6305,134: VPOKE 6308,134: VPOKE 6317,134: VPOKE 6319,134: VPOKE 6321,134: VPOKE 6323,134: VPOKE 6331,134: VPOKE 6334,134		
115	VPOKE 6337,134: VPOKE 6340,134: VPOKE 6341,134: VPOKE 6342,134: VPOKE 6347,134: VPOKE 6348,134: VPOKE 6349,134:		

116 VPOKE 6351,134:VPOKE 6352,134  
 :VPOKE 6353,134: VPOKE 6355,  
 134 VPOKE 6361,134  
 120 VPOKE 6362,134: VPOKE 6363,  
 134: VPOKE 6366,134  
 VPOKE 6369,134:VPOKE 6374,  
 134:VPOKE 6379,134:VPOKE 6384  
 ,134: VPOKE 6387,134: VPOKE  
 6388,134: VPOKE 6389,134: VPO  
 KE 6393,134: VPOKE 6398,134  
 125 VPOKE 6401,134: FOR I=6404 TO  
 6411: VPOKE I,134: NEXT: VPOK  
 E 6414,134: VPOKE 6415,134: V  
 POKE 6416,134: VPOKE 6421,134  
 : VPOKE 6425,134: VPOKE 6430,  
 134  
 130 FOR I=6433 TO 6436: VPOKE I,  
 134: NEXT: VPOKE 6440,134: VP  
 OKE 6446,134: FOR I=6453 TO  
 6459: VPOKE I,134: NEXT: VPOK  
 E 6462,134  
 135 VPOKE 6466,134: FOR I=6468 TO  
 6474: VPOKE I,134: NEXT: FOR  
 I=6478 TO 6483: VPOKE I,134:  
 NEXT: VPOKE 6487,134: FOR I=  
 6491 TO 6494: VPOKE I,134: NE  
 XT  
 140 VPOKE 6498,134:VPOKE 6501,134  
 : VPOKE 6506,134: VPOKE 6507,  
 134:VPOKE 6508,134:VPOKE 6510  
 ,134:VPOKE 6513,134: VPOKE  
 6515,134: FOR I=6517 TO 6523:  
 VPOKE I,134: NEXT: VPOKE 6525  
 ,134  
 145 VPOKE 6530,134: FOR I=6532 TO  
 6538: VPOKE I,134: NEXT: VPOK  
 E 6540,134: VPOKE 6542,134: V  
 POKE 6545,134: VPOKE 6547,134  
 : VPOKE 6548,134: VPOKE 6549,  
 134: VPOKE 6554,134: VPOKE  
 6557,134  
 150 VPOKE 6562,134:VPOKE 6563,134  
 : VPOKE 6564,134: VPOKE 6568,  
 134: FOR I=6572 TO 6577: VPOK  
 E I,134: NEXT: FOR I=6581 TO

155 6587: VPOKE I,134: NEXT: VPOK  
 E 6589,134  
 VPOKE 6594,134: FOR I=6596 TO  
 6602: VPOKE I,134: NEXT: VPOK  
 E 6609,134: VPOKE 6615,134: V  
 POKE 6619,134: VPOKE 6620,134  
 : VPOKE 6621,134  
 160 VPOKE 6626,134:VPOKE 6628,134  
 : VPOKE 6631,134: VPOKE 6634,  
 134:VPOKE 6635,134:VPOKE 6636  
 ,134: VPOKE 6639,134: VPOKE  
 6640,134: VPOKE 6641,134: FOR  
 I=6644 TO 6651: VPOKE I,134:  
 NEXT: VPOKE 6653,134  
 165 VPOKE 6658,134:VPOKE 6660,134  
 : VPOKE 6663,134: VPOKE 6668,  
 134:VPOKE 6671,134:VPOKE 6676  
 ,134: VPOKE 6680,134: VPOKE  
 6683,134: VPOKE 6685,134  
 170 FOR I=6689 TO 6692: VPOKE I,  
 134: NEXT:VPOKE 6695,134:VPOKE  
 6696,134:VPOKE 6697,134:VPOKE  
 6700,134:VPOKE 6702,134:VPOKE  
 6703,134:VPOKE 6704,134:VPOKE  
 6706,134:VPOKE 6707,134:VPOKE  
 6708,134:VPOKE 6711,134:  
 171 VPOKE 6712,134: FOR I=6715 TO  
 6718: VPOKE I,134: NEXT  
 175 VPOKE 6721,134:VPOKE 6724,134  
 : VPOKE 6729,134: VPOKE 6732,  
 134:VPOKE 6734,134:VPOKE 6736  
 ,134: VPOKE 6738,134: VPOKE  
 6743,134: VPOKE 6747,134: VPO  
 KE 6750,134  
 180 FOR I=6753 TO 6766: VPOKE I,  
 134: NEXT: FOR I=6768 TO 6782  
 : VPOKE I,134: NEXT  
 185 VPOKE 6792,134:VPOKE 6797,134  
 : VPOKE 6801,134: VPOKE 6806,  
 134: FOR I=6829 TO 6833: VPOK  
 E I,134: NEXT  
 190 VPOKE 6241,160:VPOKE 6270,160  
 : FOR I=6242 TO 6782: IF VP  
 EE K(1)=134 THEN K=RND(1): IF K<  
 .05 THEN VPOKE I,168 ELSE IF

```

192 KK.1 THEN VPOKE I,152 ELSE IF
    KK.15 THEN VPOKE I,144
    NEXT: VPOKE 6530,160: VPOKE
    6589,160: VPOKE 6278,160: VPO
    KE 6297,160
195 M$(1)="Tu_turno_": M$(2)="
    Espera_": GOSUB 1000: JA=
    1: Q=1: PUT SPRITE 0,(64,160)
    ,1: PUT SPRITE 1,(176,160),12:
    CJ(1)=134: CJ(2)=134: PJ(1)=
    6792: PJ(2)=6806

```

Editor

```

200 X$=INKEY$: C=C+1
210 IF C=6 THEN Q=1-Q: C=0
220 IF Q<>0 THEN 250
230 Q=1: K=VPEEK(PJ(JA)): IF K=32
    THEN VPOKE PJ(JA),CJ(JA) ELSE
    VPOKE PJ(JA),32
250 IF X$="" THEN 200
260 K=ASC(X$)-27: IF K<1 OR K>4 T
    HEN 200 ELSE ON K GOTO 300,
    320,340,360

```

Test de movimientos permitidos

```

300 IF VPEEK(PJ(JA)+1)=32 THEN
    200
305 VPOKE PJ(JA),CJ(JA): CJ(JA)=V
    PEEK(PJ(JA)+1): PJ(JA)=PJ(JA)
    +1: VPOKE 6909+4*JA,VPEEK(6909
    +4*JA)+8
310 GOTO 400
320 IF VPEEK(PJ(JA)-1)=32 THEN
    200
325 VPOKE PJ(JA),CJ(JA): CJ(JA)=V
    PEEK(PJ(JA)-1): PJ(JA)=PJ(JA)
    -1: VPOKE 6909+4*JA,VPEEK(6909
    +4*JA)-8
330 GOTO 400

```

```

340 IF VPEEK(PJ(JA)-32)=32 THEN
    200
345 VPOKE PJ(JA),CJ(JA): CJ(JA)=V
    PEEK(PJ(JA)-32): PJ(JA)=PJ(JA)
    -32: VPOKE 6908 + 4*JA,VPEEK(
    6908+4*JA)-8
350 GOTO 400
360 IF VPEEK(PJ(JA)+32)=32 THEN
    200
365 VPOKE PJ(JA),CJ(JA): CJ(JA)=V
    PEEK(PJ(JA)+32): PJ(JA)=PJ(JA)
    +32: VPOKE 6908 + 4*JA,VPEEK(
    6908+4*JA)+8

```

Comprobación de movimiento

```

400 K=CJ(JA): IF K=134 THEN GOSUB
    750 ELSE IF K=144 THEN GOSUB
    600 ELSE IF K=152 THEN GOSUB
    650 ELSE IF K=160 THEN GOSUB
    550 ELSE IF K=168 THEN GOSUB
    700 ELSE GOSUB 900
401 K=CJ(JA) : ON K GOSUB 750,600
    ,650,550,700,900,900,900,900
410 IF JA=1 THEN JA=2 ELSE JA=1
420 M$(JA)="Tu_turno_": IF JA=
    1 THEN M$(2)="Espera_"EL
    SE M$(1)= "Espera_"
425 GOSUB 1000
490 GOTO 200

```

Rubies

```

550 M$(JA)="RUBIES_+_750_":P(JA)
    =P(JA)+750:GOSUB 1000: CJ(JA)
    =134: VPOKE PJ(JA),134
580 PLAY"14s8m30000n4518n4014n45"
    ,"14s8m20000n3618n3114n36": F
    OR T=1 TO 1000: NEXT
590 RETURN

```



Opalos

```
600 M$(JA)="OPALOS_+_250_":P(JA)
=P(JA)+250:GOSUB 1000: CJ(JA)
=134: VPOKE PJ(JA),134
630 PLAY"14s14m3000n45": FOR T=1
TO 1000: NEXT
640 RETURN
```

Esmeraldas

```
650 M$(JA)="ESMER.+500":P(JA)=P(J
A)+500:GOSUB 1000: CJ(JA)=134
: VPOKE PJ(JA),134
680 PLAY"14s14m3000n4012n45": FOR
T=1 TO 1000: NEXT
690 RETURN
```

Serpiente

```
700 M$(JA)="SERP.-100_":P(JA)=P(J
A)-100:GOSUB 1000: CJ(JA)=134
: VPOKE PJ(JA),134
730 PLAY"164s8m59000n45n46n47n48n
49n50n51n51n50n49n48n47n46n45
n44n45n46n47n48n49n50n51n50n4
9n48n47n46n45n4413n43": FOR T
=1 TO 2000: NEXT
740 RETURN
```

Jugador en casilla normal

```
750 IF RND(1)<.7 THEN RETURN
755 IF RND(1)<.3 THEN M$(JA)="PLA
TA+50_": P(JA)=P(JA)+50: GOS
UB 1000: PLAY"16s1m777n34n50n
23","12s5m2000n40": FOR T=1 T
O 1000: NEXT: RETURN
```

```
760 IF RND(1)<.5 THEN M$(JA)="AMA
T.+20_": P(JA)=P(JA)+20: GOS
UB 1000: PLAY"14s10m500n35n30
","12s1m1000n20": FOR T=1 TO
1000: NEXT: RETURN
765 M$(JA)="ES_ORO_-50": P(JA)=P(
JA)-50: GOSUB 1000: PLAY"11s8
m2000n2": FOR T=1 TO 1000: N
EXT: RETURN
```

Fin de juego

```
900 SCREEN 1: P(JA)=P(JA)+1000
910 PLAY"18s2m30000n45n4614n4512n
4611n37","s2m3000018n37r814n3
7r211n33"
920 IF P(1)>P(2) THEN PRINT P$(1)
"_HA_GANADO" ELSE IF P(2)>P(1
) THEN PRINT P$(2)"_HA_GANADO
" ELSE PRINT"EMPATADOS"
930 FOR I=1 TO 3: PRINT: NEXT: PR
INT"Las_puntuaciones_son:": P
RINT: PRINT P$(1),P(1): PRINT
: PRINT P$(2),P(2)
980 PRINT: PRINT: PRINT
990 END
```

Puntuación y mensaje

```
1000 PRINT P$(1)":": PRINT"PUNTOS:
";P(1);TAB(12)M$(1): FOR I=1
TO 20: PRINT: NEXT: PRINT P$(
2)":": PRINT "PUNTOS: ";P(2);T
AB(11)M$(2);CHR$(11);
1010 RETURN
```

Datos de sprites

```
10000 DATA 24,60,126,126,60,36,66,
129
```

Datos de sprites pequeños		
10100	DATA	255, 213, 171, 213, 171, 213, 171, 255
10102	DATA	66, 231, 66, 0, 0, 66, 231, 66
10104	DATA	32, 112, 248, 116, 46, 31, 14, 4
10106	DATA	60, 126, 255, 255, 255, 255, 126, 60
10108	DATA	2, 7, 12, 76, 140, 134, 195, 126
10110	DATA	1, 3, 7, 15, 31, 63, 127, 255
10112	DATA	128, 192, 224, 240, 248, 252, 254, 255
10114	DATA	255, 127, 63, 31, 15, 7, 3, 1
10116	DATA	255, 254, 252, 248, 240, 224, 192, 128

590	=	143	755	=	11653	10000	=	1570
600	=	5876	760	=	11516	10100	=	1885
630	=	2562	765	=	6628	10102	=	1390
640	=	143	900	=	1718	10104	=	1535
650	=	5919	910	=	5112	10106	=	1775
680	=	3007	920	=	6840	10108	=	1514
690	=	143	930	=	6033	10110	=	1326
700	=	5692	980	=	561	10112	=	1891
730	=	12076	990	=	129	10114	=	1343
740	=	143	1000	=	8585	10116	=	1883
750	=	1506	1010	=	143	TOTAL	=	437289

## Diseño de los sprites



JUGADOR

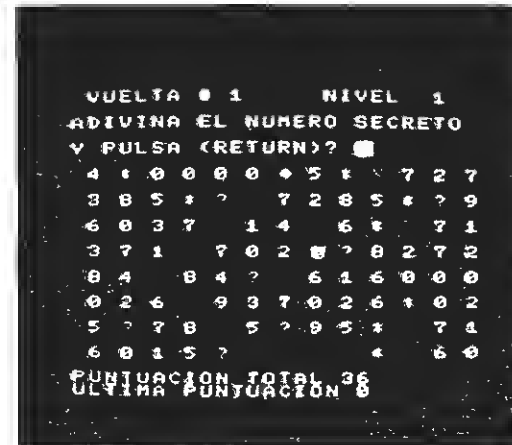


## Tabla de claves

5	=	1236	125	=	6994	250	=	1032
7	=	1940	130	=	5901	260	=	4656
10	=	4362	135	=	7883	300	=	2059
15	=	8894	140	=	9606	305	=	8509
20	=	1949	145	=	10499	310	=	537
25	=	4886	150	=	8947	320	=	2059
30	=	2951	155	=	7551	325	=	8511
35	=	3170	160	=	12260	330	=	537
40	=	2756	165	=	6289	340	=	2044
45	=	6096	170	=	12810	345	=	8518
47	=	3166	171	=	1665	350	=	537
50	=	7024	175	=	7800	360	=	2044
55	=	7081	180	=	3771	365	=	8516
57	=	3087	185	=	5077	400	=	9647
60	=	2297	190	=	11233	401	=	3568
100	=	1772	192	=	3109	410	=	1829
105	=	5850	195	=	13894	420	=	6786
110	=	6446	200	=	1325	425	=	397
115	=	11402	210	=	1793	490	=	593
116	=	2213	220	=	1220	550	=	5962
120	=	8095	230	=	4786	580	=	5622

# Minotauro Mastermind Jr.

25



Tipo de juego:

JUEGO EDUCATIVO

Este es un juego educativo, versión del "Minotauro Mastermind", en el que se han simplificado las pistas para ayudar a los jóvenes jugadores a que encuentren el número secreto entre 10 y 99.

Tu puntuación es la suma de los números sobre los que te posas; por eso son mejores las puntuaciones bajas. Cuando alcanzas la salida en lo alto del bloque tienes una oportunidad de averiguar el número secreto. Los símbolos no numéricos tienen distintos efectos cuando pasas por ellos:

- X Te da una pista.
- ? Tiene un valor aleatorio.
- \* Altera la puntuación aleatoriamente.

La situación se indica mediante un punto intermitente que controlas con:

- |                       |                      |
|-----------------------|----------------------|
| <P> Arriba.           | <:> Abajo/derecha.   |
| <Ø> Arriba/izquierda. | <a> Derecha.         |
| <:> Abajo.            | <L> Abajo/izquierda. |
| <O> Izquierda.        | <T> Atrapado.        |
| <-> Arriba/derecha.   |                      |

## SUGERENCIAS AL PROGRAMA

Puedes modificar la complejidad del juego añadiéndole varios niveles de dificultad. Una posibilidad es simplificar el cuadro que hay en pantalla (líneas 100-160), de forma que haya menos números y símbolos.

Igual que en el juego original, puedes añadir más pistas (líneas 650-688) o cambiar las existentes.

### Variables del programa

ND	Nivel de dificultad.
V	Número de vuelta.
NS	Número secreto.
D1, D2	Dígitos del número secreto.
PJ	Posición jugador.
M	Posición minotauro.
UP, TP	Ultima puntuación; total puntos.
NA	Número aleatorio.
CB, J	Carácter debajo del minotauro: jugador.
FJ, CJ	Fila y columna del jugador.
NP	Número de pistas.
T(I)	¿Pista utilizada?
G1, G2	Dígitos supuestos por el jugador.
MC, MF	Columna y fila del minotauro.

		Inicialización	
	4	GOSUB 20000	
	5	TM=1: INTERVAL ON: ON INTERVA	
		L=10 GOSUB 7	
	6	GOTO 10	
	7	TM=TM*2: RETURN	
	10	KEY OFF: SCREEN 1: CLS: COLOR	
		15,1,1: PRINT"____MINOTAURO_	
		MASTER_MIND"	
	15	PRINT: PRINT: PRINT: PRINT "_	
		NIVEL_DE_DIFICULTAD_(1-9)";	
	20	ND=VAL(INKEY\$): IF ND<1 OR ND	
		>9 THEN 20 ELSE PRINT ND	
	22	FOR I=1088 TO 1344 STEP 64: F	
		OR J=0 TO 7: READ Q: VPOKE I+	
		J,Q: NEXT: NEXT	

	23	VPOKE 8209,49: VPOKE 8210,97:	
		VPOKE 8211,177: VPOKE 8212,	
		145: VPOKE 8213,129	
	25	FOR I=1 TO 8: PRINT: NEXT: PR	
		INT "PULSA UNA TECLA PARA EMP	
		EZAR"	
	30	IF INKEY\$="" THEN 30	
	35	INTERVAL OFF: IF TM>1000 THEN	
		TM=TM/3: GOTO 35	
	40	TP=0: NP=0: FOR I=1 TO 6: T(I	
		)=0: NEXT: V=1: NS=INT(89*TM/	
		1000+10): N=NS*100: D1=INT(NS	
		/10): D2=INT(NS-D1*10)	
		Nueva jugada	
	100	CLS: PJ=6799: M=6607	
	110	FOR I=6336 TO 6784 STEP 64: F	
		OR J=3 TO 27 STEP 2: GOSUB	
		900: K1=INT(NA*13): IF K1<10	
		THEN VPOKE I+J,K1+48: GOTO	
		130	
	115	IF K1=10 THEN VPOKE I+J,144:	
		GOTO 130	
	120	IF K1=11 THEN VPOKE I+J,152:	
		GOTO 130	
	125	VPOKE I+J,160	
	130	NEXT: NEXT	
	140	VPOKE PJ,48: UP=0: CM=VPEEK(M	
		): VPOKE M,136: VPOKE 6351,	
		168: CJ=48	
	150	PRINT "_VUELTA_#";V;"____NIVE	
		L_";ND; CHR\$(11);	
	160	GOSUB 1000	
		Editor	
	200	C=0: Q=1	
	205	X\$=INKEY\$: C=C+1	
	210	IF C=7 THEN Q=1-Q: C=0	
	215	IF Q<>0 THEN 230	
	220	IF VPEEK(PJ)=255 THEN VPOKE P	

```

J,J: Q=1: GOTO 230
225 VPOKE PJ,255: Q=1
230 IF X$="" THEN 205
240 GOSUB 950
242 IF VPEEK(PJ)=255 THEN VPOKE P
    J,J
245 IF X$="P" OR X$="p" THEN 300
250 IF X$="O" OR X$="o" THEN 310
255 IF X$="e" OR X$="2" THEN 320
260 IF X$=";" THEN 330
265 IF X$="-" THEN 340
270 IF X$="0" THEN 350
275 IF X$="L" OR X$="1" THEN 360
280 IF X$=":" THEN 370
285 IF X$="T" OR X$="t" THEN 380
290 GOTO 205

```

Test de movimientos permitidos

```

300 IF VPEEK(PJ-64)=32 THEN 205
302 UP=VPEEK(PJ)-48
304 VPOKE PJ,32: PJ=PJ-64
305 GOTO 400
310 IF VPEEK(PJ-2)=32 THEN 205
312 UP=VPEEK(PJ)-48
314 VPOKE PJ,32: PJ=PJ-2
315 GOTO 400
320 IF VPEEK(PJ+2)=32 THEN 205
322 UP=VPEEK(PJ)-48
324 VPOKE PJ,32: PJ=PJ+2
325 GOTO 400
330 IF VPEEK(PJ+64)=32 THEN 205
332 UP=VPEEK(PJ)-48
334 VPOKE PJ,32: PJ=PJ+64
335 GOTO 400
340 IF VPEEK(PJ-62)=32 THEN 205
342 UP=VPEEK(PJ)-48
344 VPOKE PJ,32: PJ=PJ-62
345 GOTO 400
350 IF VPEEK(PJ-66)=32 THEN 205
352 UP=VPEEK(PJ)-48
354 VPOKE PJ,32: PJ=PJ-66
355 GOTO 400

```

```

360 IF VPEEK(PJ+62)=32 THEN 205
362 UP=VPEEK(PJ)-48
364 VPOKE PJ,32: PJ=PJ+62
365 GOTO 400
370 IF VPEEK(PJ+66)=32 THEN 205
372 UP=VPEEK(PJ)-48
374 VPOKE PJ,32: PJ=PJ+66
375 GOTO 400
380 FJ=INT(PJ/32): CJ=PJ-FJ*32: G
    OSUB 500: IF PJ<>M THEN 380
385 GOTO 800

```

Comprobación de movimiento

```

400 J=VPEEK(PJ): FJ=INT(PJ/32): C
    J=PJ-FJ*32
402 IF (UP>9) OR (UP<0) THEN UP=0
405 K1=VPEEK(PJ): IF K1=152 THEN
    GOSUB 600
410 IF K1=144 THEN GOSUB 650
415 IF K1=160 THEN GOSUB 750
420 IF K1=168 THEN 2000
422 K1=VPEEK(PJ)-48: IF K1<0 OR K
    1>9 THEN K1=0
425 IF U+K1>=10-ND THEN GOSUB 500
430 TP=TP+K1: GOSUB 1000
435 IF PJ=M THEN 800
440 GOTO 200

```

Movimiento del minotauro

```

500 VPOKE 8198,31: MF=INT(M/32):
    MC=M-MF*32
502 IF MC<>J THEN 508
504 IF M>FJ THEN MF=MF-2: GOTO
    540
506 MF=MF+2: GOTO 540
508 IF MF<>FJ THEN 515
510 IF MC>J THEN MC=MC-2: GOTO
    540
512 MC=MC+2: GOTO 540

```

```

515 IF M>PJ THEN 522
517 IF MC<PJ THEN MC=MC+2: GOTO
540
520 MC=MC-2: GOTO 540
522 MF=MF-2
540 K=32*MF+MC: VPOKE 8198,241:IF
VPEEK(K)=136 THEN RETURN
542 VPOKE M,CB
544 M=K: CB=VPEEK(M)
546 VPOKE M,136
550 PLAY "n2"
570 RETURN

```

Jugador sobre (?)

```

600 GOSUB 900: I=INT(NA*10): VPOKE
PJ,1+48
610 J=1+48: RETURN

```

Jugador sobre (X)

```

650 VPOKE PJ,48
652 IF NP>=6 THEN BEEP: GOTO 670
654 GOSUB 900: K2=INT(NA*6+1): IF
T(K2)=1 THEN 654
656 T(K2)=1: NP=NP+1: BEEP
658 ON K2 GOTO 660,665,670,675,
680,685
660 PRINT: PRINT: PRINT"La suma d
e los digitos del":PRINT"nume
ro secreto es ";
662 PRINT D1+D2: PRINT CHR$(11);:
RETURN
665 PRINT: PRINT: PRINT"El numero
secreto es ";
666 IF NS>50 THEN PRINT"Mayor qu
e 50": GOTO 668
667 PRINT"Menor o igual que 50"
668 PRINT CHR$(11);: RETURN
670 GOSUB 900:K=INT(NA*15)+NS:J=K
-15: IF J<0 THEN J=0

```

```

672 IF K>99 THEN K=99
673 PRINT: PRINT: PRINT"El numero
esta entre";J: PRINT"y";K
674 PRINT CHR$(11);: RETURN
675 PRINT: PRINT: PRINT"El primer
digito es"
676 IF D1<5 THEN PRINT"menor que
5":GOTO 678
677 PRINT"mayor o igual que 5"
678 PRINT CHR$(11);: RETURN
680 PRINT: PRINT: PRINT"El segund
o digito es"
681 IF D2>5 THEN PRINT "menor que
5": GOTO 683
682 PRINT"mayor o igual que 5"
683 PRINT CHR$(11);: RETURN
685 PRINT: PRINT: PRINT"El produc
to de los dos": PRINT"digitos
es";
688 PRINTD1*D2: CHR$(11);: RETURN

```

Jugador sobre (\*)

```

750 VPOKE PJ,48
752 IF T<11 THEN RETURN
754 GOSUB 900: I=INT(NA*100): IF
1>50 THEN T=T-10: PLAY"L64N10
N8N6N4": RETURN
756 IF 1<10 THEN T=T+50: PLAY"L32
N10N8N6N4N10N8N6N4": RETURN
760 T=T+10: PLAY"L64N10N8N6N4": R
ETURN

```

Fin de juego

```

800 CLS: IF M=PJ THEN 820
805 FOR I=1 TO 8: PRINTTAB(2*I)"E
NHORABUENA": PRINT: NEXT:PRIN
T
810 PRINT"HAS VENCIDO AL MINOTAUR
O": PRINT"TU PUNTUACION ES ";
TP

```

```

815 PRINT"EN";V;"_VUELTAS": GOTO
840
820 FOR I=1 TO 5: PRINT TAB(2*I)"
VENCIO_EL_MINOTAURO": PRINT:
NEXT: PRINT
825 PRINT"El_numero_secreto_era_"
;NS
840 PRINT: PRINT"PULSA_UNA_TECLA_
PARA": PRINT"JUGAR_OTRA_VEZ"
845 X$=INKEY$: IF X$="" THEN 845
846 RUN 5

```

Número aleatorio

```

900 NA=(9999*NA+N) MOD 5997!: NA=
NA/5997!
910 IF NA<.2 THEN N=NA*10000+1
920 RETURN

```

Mensaje en blanco

```

950 FOR I=6208 TO 6271: VPOKE I,
32: NEXT
970 RETURN

```

Puntuación

```

1000 FOR I=1 TO 22: PRINT: NEXT: P
RINT"PUNTUACION_TOTAL";TP: PR
INT"ULTIMA_PUNTUACION";UP;CHR
$(11);
1010 RETURN

```

Adivina el número secreto

```

2000 PRINT: PRINT: PRINT"ADIVINA_E
L_NUMERO_SECRETO": PRINT
2005 INPUT"Y_PULSA_(RETURN)";K$

```

```

2010 I=VAL(K$): IF I=NS THEN 800
2015 G1=INT(I/10): G2=I-G1*10
2020 CLS: PRINT: IF D1=G1 THEN PRI
NT"El_primer_digito_es_correc
to":PRINT
2025 IF D2=G2 THEN PRINT"El_segund
o_digito_es_correcto":PRINT
2035 IF D1<>G1 AND D2<>G2 THEN PRI
NT"Ningun_digito_es_correcto"
2040 FOR Z=1 TO 2000: NEXT
2045 V=V+1: GOTO 100

```

Datos de los sprites pequeños

```

10000 DATA 165,126,219,126,102,126,
82,60
10002 DATA 136,136,80,32,80,136,136
,0
10004 DATA 112,136,8,16,32,0,32,0
10006 DATA 32,168,112,32,112,168,32
,0
10008 DATA 16,56,124,254,124,56,16,
0

```

Instrucciones

```

20000 CLS: INPUT"QUIERES_INSTRUCCION
ES_(S/N)?";X$: IF X$="N" OR X
$="n" THEN 20010
20001 CLS: PRINT"PARA_MOVERTE_EN_EL
LABERINTO_PULSA_:"
20005 PRINT:PRINT"__P-p__ARRIBA":
PRINT"__O-o__IZQUIERDA": PRI
NT"__@-2__DERECHA": PRINT"__
;__ABAJO": PRINT"__~__ARRIBA-
DERECHA"
20006 PRINT"__0__ARRIBA-IZQUIERDA
": PRINT"__L-l__ABAJO-IZQUIE
RDA": PRINT"__:__ABAJO-DER
ECHA": PRINT"__T-t__ATRAPADO
"
20007 PRINT:PRINT:PRINT"SI_PISAS_:"

```

○		: PRINT: PRINT"X.....TENDRAS UNA_PISTA": PRINT"?.....SUMAR AS_PUNTUACION.....ALEAT ORIA": PRINT"*.....ALTERARA_T U_PUNTUACION.....ALEATORIAM ENTE"	○
○			○
○	20008	PRINT: PRINT"PULSA UNA TECLA PARA SEGUIR"	○
○	20009	IF INKEY\$="" THEN 20009	○
○	20010	RETURN	○

## Tabla de claves

4	= 234	242	= 2045	345	= 537
5	= 3376	245	= 1850	350	= 1850
6	= 403	250	= 1823	352	= 1391
7	= 1035	255	= 1792	354	= 1389
10	= 3886	260	= 966	355	= 537
15	= 2866	265	= 970	360	= 1838
20	= 3352	270	= 977	362	= 1391
22	= 3640	275	= 1886	364	= 1372
23	= 2630	280	= 1013	365	= 537
25	= 4004	285	= 1914	370	= 1850
30	= 969	290	= 594	372	= 1391
35	= 3770	300	= 1848	374	= 1388
40	= 10803	302	= 1391	375	= 537
100	= 647	304	= 1389	380	= 4613
110	= 7895	305	= 537	385	= 391
115	= 2238	310	= 1784	400	= 3786
120	= 2239	312	= 1391	402	= 2141
125	= 822	314	= 1327	405	= 2292
130	= 320	315	= 537	410	= 1197
140	= 4118	320	= 1784	415	= 1313
150	= 2880	322	= 1391	420	= 1162
160	= 397	324	= 1326	422	= 3276
200	= 734	325	= 537	425	= 2151
205	= 1325	330	= 1848	430	= 1426
210	= 1798	332	= 1391	435	= 860
215	= 1208	334	= 1388	440	= 593
220	= 3244	335	= 537	500	= 2954
225	= 1087	340	= 1838	502	= 1347
230	= 1036	342	= 1391	504	= 2196
240	= 345	344	= 1373	506	= 1297

508	= 1176	674	= 932	970	= 143
510	= 2185	675	= 2657	1000	= 6679
512	= 1294	676	= 2793	1010	= 143
515	= 878	677	= 2062	2000	= 2930
517	= 2294	678	= 932	2005	= 1653
520	= 1293	680	= 2779	2010	= 1877
522	= 803	681	= 2783	2015	= 2414
540	= 3393	682	= 2062	2020	= 4758
542	= 467	683	= 932	2025	= 4459
544	= 1407	685	= 4380	2035	= 5072
546	= 481	688	= 1514	2040	= 1161
550	= 431	750	= 469	2045	= 1258
570	= 143	752	= 863	10000	= 1770
600	= 2555	754	= 4880	10002	= 1572
610	= 908	756	= 3610	10004	= 1340
650	= 469	760	= 2120	10006	= 1567
652	= 1847	800	= 1143	10008	= 1517
654	= 3233	805	= 3305	20000	= 5176
656	= 1753	810	= 4459	20001	= 3468
658	= 2198	815	= 1928	20005	= 9275
660	= 5728	820	= 4060	20006	= 8483
662	= 1663	825	= 2714	20007	= 16353
665	= 2939	840	= 3734	20008	= 2636
666	= 2917	845	= 1602	20009	= 1050
667	= 2133	846	= 164	20010	= 143
668	= 932	900	= 3262		
670	= 4096	910	= 1952	TOTAL=	370203
672	= 1233	920	= 143		
673	= 3497	950	= 1584		



# Treinta y una 26



Tipo de juego:

JUEGO DE MESA

En este juego, el MSX, te desafía a una mano de cartas; se suman los puntos de cada carta y gana el que más cerca esté de 31, pero sin sobrepasarlas, ya que, si lo haces, pierdes (o el MSX, si es éste quien la sobrepasa).

Los puntos de las cartas son:

Rey	13
Dama	12
Jota	11
As	1

Resto de cartas, su número.

## SUGERENCIAS AL PROGRAMA

El MSX actúa de forma sencilla, se planta cuando alcanza 25; pero si, aun así, tú la superas, cambia de táctica memorizando las cartas utilizadas y calculando las probabilidades de la carta siguiente. Puedes incluso hacer un poco de trampa, mirando un instante la siguiente carta (con las líneas 300-310).

### Variables del programa

M1\$, M2\$	Mensajes.
MC(I, J)	Matriz de cartas.
BC(I)	Baraja de cartas.
SC	Siguiente carta de la baraja.
AB	Apuestas de esta baza.
BC	Balance de la cuenta.
PJ	Puntos jugador.
PO	Puntos ordenados.
LC	Localización de la carta
D	Descarte.
CO	Número de cartas del ordenador.
CJ	Número de cartas del jugador.

		Inicialización	
10	KEY OFF: SCREEN 1,2: COLOR 15,4,7		
15	FOR I=1 TO 32: READ Q: A\$=A\$+CHR\$(Q): NEXT: SPRITE\$(0)=A\$: A\$=""		
20	FOR I=1 TO 32: READ Q: A\$=A\$+CHR\$(Q): NEXT: SPRITE\$(1)=A\$: A\$=""		
25	FOR I=1 TO 32: READ Q: A\$=A\$+CHR\$(Q): NEXT: SPRITE\$(2)=A\$: A\$=""		
30	FOR I=1072 TO 1143: READ Q: VPOKE I,Q: NEXT		
35	PRINT: PRINT: PRINT".....REINTA_Y_UNA": VPOKE 6152,134:FOR I=6153 TO 6167: VPOKEI,135: NEXT:VPOKE 6168,136: VPO		

		KE 6200,137: VPOKE 6232,137: VPOKE 6184,141: VPOKE 6216,141	
36	FOR I=6281 TO 6295: VPOKE I,139: NEXT		
40	VPOKE 6264,137: VPOKE 6296,138: VPOKE 6248,141: VPOKE 6280,140		
45	FOR I=384 TO 463: K=VPEEK(I): VPOKE I,INT(K/4): NEXT		
50	FOR I=1 TO 6: PRINT: NEXT: C=100: PRINT" Empiezas_con_100_pesetas": PRINT: PRINT".....Pulsa_una_tecla"		
55	IF INKEY\$="" THEN I=RND(1): GOTO 55		
60	CLS: DIM MC(52,2),BC(52)		
65	FOR I=1 TO 4: FOR J=1 TO 13: K=(I-1)*13+J: MC(K,1)=J: MC(K,2)=I: NEXT: NEXT: FOR I=1 TO 52: BC(I)=I: NEXT		
75	C=100: SC=1		
80	PRINT TAB(12)"Tu_mano"		
90	PRINT CHR\$(11);:M1\$="Por_favor_espera_que_baraje": M2\$="1a s_cartas.": GOSUB 1000: GOSUB 1100: GOSUB 1200: IF SC=1 THEN GOSUB 400: GOSUB 300: GOSUB 200: GOSUB 200		
		Control	
100	IF ((ST) AND 1)=1 THEN 102 ELSE GOSUB 350		
102	IF VPEEK(6156)=77 THEN 130 ELSE IF ((ST) AND 1)<>1 THEN 110		
105	FOR I=1 TO 5: BEEP: PRINT CHR\$(11) TAB(20): FOR I2=1 TO 300: NEXT: PRINT CHR\$(11) TAB(10)"Mano_del_MSX" CHR\$(11);: NEXT: FOR I=6176 TO 6223: VPOKE I,32: NEXT		

```

106 FOR I=0 TO 2: PUT SPRITE I,
    (100,200): NEXT
107 FOR I1=1 TO 10: IF I1<7 THEN
    LC=6179+3*I1 ELSE LC=6278+3*(
    I1-7)
108 D=CH(I1): GOSUB 250: NEXT: FO
    R I=1 TO 15: PRINT TAB(15)"Pu
    ntos: ";PO;CHR$(I1)
110 IF ((ST) AND 1)=1 THEN 130
120 M1$="Tu_siguiente_carta_
    _":M2$="
    _":GOSUB 1000
125 GOSUB 200: IF PJ>31 THEN 500
130 IF((ST) AND 2)=2 THEN 135 ELS
    E M1$="Siguiente_carta_del_MS
    X_": M2$="
    _":GOSUB 1000: FOR T=1
    TO 1500: NEXT: GOSUB 300
131 IF (PO>24 AND PO<32) THEN ST=
    ((ST) OR 2) ELSE IF PO>31 THE
    N 600
135 IF ST=3 THEN 450
140 GOTO 100

```

Mano del jugador

```

200 D=BC(SC): SC=SC+1
205 PJ=PJ+MC(D,1): GOSUB 1100: CJ
    =CJ+1
210 IF CJ<7 THEN LC=6179+3*CJ ELS
    E LC=6278+3*(CJ-7)
215 GOSUB 250
240 RETURN

```

Dibujar carta

```

250 VPOKE LC,134: FOR I=LC+1 TO L
    C+5: VPOKE I,135: NEXT: VPOKE
    LC+6,136: FOR I=LC+33 TO LC+
    289 STEP 32: FOR J=0 TO 4: VP
    OKE I-1,141: VPOKE I+5,137: V
    POKE I+J,142: NEXT: NEXT

```

```

255 VPOKE LC+320,140: FOR I=LC+
    321 TO LC+325: VPOKE I,139:
    NEXT: VPOKE LC+326,138
260 K1=MC(D,2): IF K1=1 THEN K1=3
    ELSE IF K1=2 THEN K1=4 ELSE
    IF K1=3 THEN K1=5 ELSE K1=6
265 K2=MC(D,1): ON K2 GOTO 270,
    272,274,276,278,280,282,284,
    286,288,290,292,294
270 VPOKE LC+33,65: VPOKE LC+163,
    K1: VPOKE LC+293,65: RETURN
272 VPOKE LC+33,50: VPOKE LC+67,
    K1: VPOKE LC+259,K1: VPOKE LC
    +293,50: RETURN
274 VPOKE LC+33,51: VPOKE LC+67,
    K1: VPOKE LC+163,K1: VPOKE LC
    +259,K1: VPOKE LC+293,51:
    RETURN
276 VPOKE LC+33,52: VPOKE LC+293,
    52: VPOKE LC+66,K1: VPOKE LC+
    68,K1: VPOKE LC+258,K1: VPOKE
    LC+260,K1: RETURN
278 VPOKE LC+33,53: VPOKE LC+293,
    53: VPOKE LC+66,K1: VPOKE LC+
    68,K1: VPOKE LC+258,K1: VPOKE
    LC+260,K1: VPOKE LC+163,K1:
    RETURN
280 VPOKE LC+33,54: VPOKE LC+293,
    54: VPOKE LC+67,K1: VPOKE LC+
    130,K1: VPOKE LC+132,K1: VPOK
    E LC+194,K1: VPOKE LC+196,K1:
    VPOKE LC+259,K1: RETURN
282 VPOKE LC+33,55: VPOKE LC+293,
    55: VPOKE LC+67,K1: VPOKE LC+
    130,K1: VPOKE LC+132,K1: VPOK
    E LC+194,K1: VPOKE LC+196,K1:
    VPOKE LC+258,K1: VPOKE LC+260
    ,K1: RETURN
284 VPOKE LC+33,56: VPOKE LC+293,
    56: VPOKE LC+66,K1: VPOKE LC+
    68,K1: VPOKE LC+130,K1: VPOKE
    LC+132,K1: VPOKE LC+194,K1: V
    POKE LC+196,K1: VPOKE LC+258,
    K1: VPOKE LC+260,K1: RETURN
286 VPOKE LC+33,57: VPOKE LC+293,

```

Mano del MSX

```
300      CO=CO+1
```

```

305 D=BC(SC): SC=SC+1: PO=PO+MC(D
    ,1): CH(CO)=D
307 IF((ST AND 1)<>1 THEN 310 EL
    SE IF CO<7 THEN LC=6179+3*CO
    ELSE LC=6278+3*(CO-7)
308 GOSUB 250: FOR I=1TO 15: PRIN
    T: NEXT: PRINT TAB(15)"Puntos
    :";PO; CHR$(11);
310 RETURN

```

Apuesta

```

350      GOSUB 1300: FOR I=1 TO 18: PR
INT: NEXT: IF AB<>0 THEN 355
ELSE 360
355      INPUT "Cambias tu apuesta (S/N
)"; X$: IF X$="n" OR X$="N" TH
EN 365 ELSE INPUT "Cuanto mas
"; X$: IF VAL(X$)>1000 THEN PRI
NT CHR$(11);: GOTO 350 ELSE I
F VAL(X$)<0 THEN PRINT CHR$(
11);: GOTO 350 ELSE EL=1
356      IF EL=1 THEN AB=AB+VAL(X$): C
=C-VAL(X$): GOTO 365
360      INPUT "Apuesta inicial"; X$: IF
VAL(X$)>1000 THEN PRINT CHR$(
11);: GOTO 350 ELSE AB=VAL(X$
): C=C-VAL(X$): GOTO 365
365      IF ((ST) AND 1)=0 THEN INPUT "O
tra carta (S/N)"; X$: IF X$<>"
S" AND X$<>"s" THEN ST=(ST)
OR 1
370      PRINT CHR$(11);: GOSUB 1300:
GOSUB 1200: EL=0: RETURN

```

## Barajar cartas

```
400   FOR I=1 TO 100: J=INT(RND(1)*
      52+1): K=INT(RND(1)*52+1): SW
      AP BC(J),BC(K)
410   NEXT
440   RETURN
```

Descubrir las cartas

```
450 GOSUB 1300: FOR I=1 TO 18: PR
INT: NEXT: PRINT"El_Msx_tiene
";PO;"puntos": PRINT: K4=31-P
0: K5=31-PJ: IF K4=K5 THEN PR
INT"-Estais_empatados!": PRIN
T"-Otra_mano!": C=C+AB: GOSUB
460: CS: SC=1: GOTO 520
460 FOR T=1 TO 2000: NEXT: IF K4>
K5 THEN 600 ELSE 500
```

Pierde el jugador

```
500 PLAY"m59000s816n2r6n3r6n2": G
OSUB 1300: PRINT CHR$(11): F
OR I=1 TO 17: PRINT: NEXT: PR
INT"Has_perdido_esta_mano": P
RINT"Tus_puntos_son":PJ: ",_l
os_del_Msx:":PO: INPUT"Otra_
mano_(S/N)":X$
505 IF X$="S" OR X$="s" THEN IF S
C<25 THEN PRINT"Bien_la_misma
_baraja" FOR T=1 TO 2000: NEX
T: CLS: GOTO 520 ELSE PRINT"B
ien_otra_baraja": FOR T=1 TO
2000: NEXT: CLS: SC=1: GOTO
520
510 GOTO 900
520 AB=0: PJ=0: ST=0: PO=0: CO=0:
CJ=0: FOR I=0 TO 2: PUT SPRIT
E I,(100,200): NEXT: GOTO 80
```

Gana el jugador

```
600 PLAY"11s14m3000n44","11s14m20
00n27": GOSUB 1300: PRINT CHR
$(11): FOR I=1 TO 17: PRINT:
NEXT: PRINT"Has_ganado_esta_m
ano": PRINT"Tus_puntos_son":P
```

```
J: ",_los_del_Msx:":PO: INPU
T"Otra_mano_(S/N)":X$
605 IF X$="S" OR X$="s" THEN IF S
C<25 THEN PRINT"Bien_la_misma
_baraja": FOR T=1 TO 2000: NE
XT: CLS: GOTO 620 ELSE PRINT"
Bien_otra_baraja": FOR T=1 T
O 2000: NEXT: CLS: SC=1: GOTO
620
610 C=C+2*AB: GOTO 900
620 C=C+2*AB:AB=0: PJ=0: ST=0: PO
=0: CO=0: CJ=0: FOR I=0 TO 2:
PUT SPRITE I,(100,200): NEXT:
GOTO 80
```

Fin de juego

```
900 SCREEN 1: IF SC=0 THEN PRINT"
Debes_":PRINT USING"#####.##
":-C:PRINT"_Pesetas": PRINT:
PRINT"Los_cobradores_te_visit
aran": PRINT"dentro_de_unos_d
ias.": END
910 PRINT"Has_ganado ": PRINT US
ING "#####.##":C:PRINT"_Pese
tas": PRINT: PRINT"--Apuesto_
a_que_no_lo_haces": PRINT"dos
_veces!!": END
```

Imprimir mensaje

```
1000 FOR I=1 TO 18: PRINT: NEXT: P
RINT M1$: PRINT M2$:CHR$(11):
RETURN
1010
```

Puntuación

```
1100 FOR I=1 TO 15: PRINT: NEXT: P
RINT TAB(15)"Puntos:":PJ:CHR$
(11):
1110 RETURN
```

Imprimir cuenta

```
1200 FOR I=1 TO 23: PRINT: NEXT: P
      RINT TAB(4)"Tu_cuentas.es";: P
      RINT USING "#####.##";C;: PR
      INT CHR$(11);
1210 RETURN
```

Area de mensaje en blanco

```
1300 FOR I=6722 TO 6879: VPOKE I,
      32: NEXT
1310 RETURN
```

Datos de sprites

```
10000 DATA 255,127,42,31,19,29,9,16
        ,32,56,16,28,16,16,17,14,255,
        254,172,248,248,248,188,136,
        116,92,84,108,106,218,151,7
10010 DATA 0,5,11,15,31,62,60,121,
        120,120,120,124,124,124,62,31
        ,128,80,232,248,28,12,14,39,7
        ,7,199,15,207,15,30,252
10020 DATA 4,10,15,5,7,5,10,10,10,
        10,21,21,26,54,73,48,68,170,
        254,84,252,72,136,152,132,130
        ,28,8,24,8,8,240
```

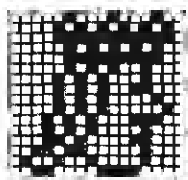
Datos de sprites pequeños

```
10100 DATA 0,0,0,0,3,7,15,15
10102 DATA 0,0,0,0,255,255,255,255
10104 DATA 0,0,0,0,192,224,240,240
10106 DATA 240,240,240,240,240,240,
        240,240
10108 DATA 240,240,224,192,0,0,0,0
10110 DATA 255,255,255,255,0,0,0,0
10112 DATA 15,15,7,3,0,0,0,0
10114 DATA 15,15,15,15,15,15,15,15
10116 DATA 255,255,255,255,255,255,
        255,255
```

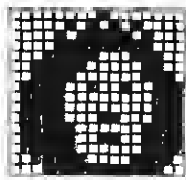
## Tabla de claves

10	= 1215	250	= 11749	440	= 143
15	= 3995	255	= 4868	450	= 16345
20	= 3996	260	= 5790	460	= 2802
25	= 3997	265	= 3980	500	= 15868
30	= 1931	270	= 3099	505	= 13717
35	= 10459	272	= 4014	510	= 523
36	= 2095	274	= 5330	520	= 6135
40	= 2984	276	= 6514	600	= 17376
45	= 3631	278	= 7843	605	= 13880
50	= 6973	280	= 9481	610	= 1640
55	= 2290	282	= 10867	620	= 7498
60	= 1128	284	= 12206	900	= 11808
65	= 7614	286	= 13803	910	= 9735
75	= 908	288	= 12745	1000	= 2670
80	= 1212	289	= 4394	1010	= 143
90	= 11488	290	= 15167	1100	= 3518
95	= 1071	291	= 1043	1110	= 143
100	= 1925	292	= 15182	1200	= 5275
102	= 3452	293	= 1043	1210	= 143
105	= 10636	294	= 15168	1300	= 1682
106	= 1926	295	= 1043	1310	= 143
107	= 5030	298	= 143	10000	= 9134
108	= 4597	300	= 805	10010	= 8616
110	= 1417	305	= 3916	10020	= 7679
120	= 5587	307	= 6176	10100	= 1033
125	= 1520	308	= 4099	10102	= 1384
130	= 10844	310	= 143	10104	= 1375
131	= 4117	350	= 3166	10106	= 1864
135	= 1017	355	= 16662	10108	= 1403
140	= 489	356	= 4279	10110	= 1416
200	= 1601	360	= 9239	10112	= 1025
205	= 2497	365	= 6880	10114	= 1404
210	= 3997	370	= 1640	10116	= 1888
215	= 410	400	= 6143		
240	= 143	410	= 131	TOTAL=	526351

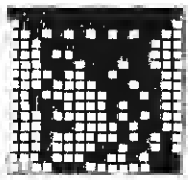
Diseño de los sprites



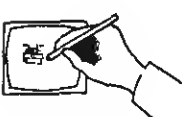
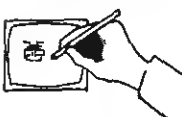
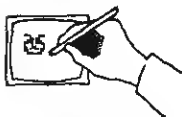
REY



REINA



JOTA



# Minotauro Mastermind

# 27



Tipo de juego:

EDUCATIVO

Este es un juego estratégico que prueba la capacidad mental, incluso de los más experimentados jugadores. El reto es encontrar un número entre 0 y 1000, abriéndote paso a través de un muro de número y símbolos sin ser atrapado por el minotauro.

Tu puntuación es la suma de los dígitos sobre los que te poses; por lo tanto, cuanto más bajos sean los puntos, mejor. Cuando consigas salir por la parte superior del bloque, tienes una oportunidad para averiguar el número.

Los efectos de los símbolos no numéricos son:

- X Te da una pista.
- ? Tiene un valor aleatorio.
- \* Altera tu puntuación aleatoriamente.



## SUGERENCIAS AL PROGRAMA

Este juego se puede complicar añadiendo más pistas y símbolos diferentes (líneas 650-700).

Tu posición la indica un punto intermitente, que controlas con:

<P> Arriba.	<: > Abajo/derecha.
<0> Arriba/izquierda.	<@> Derecha.
<; > Abajo.	<L> Abajo/izquierda.
<O> Izquierda.	<T> Atrapado.
<- > Arriba/derecha.	

## Variables del programa

ND	Nivel de dificultad.
V, UP, T	Número vuelta; última puntuación; total.
NS	Número secreto.
D1, D2, D3	Dígitos del número secreto.
PJ	Posición jugador.
PM	Posición minotauro.
C, Q	Efectos luminosos.
TP	Tiempo.
NA, N	Número aleatorio; semilla del número aleatorio.
CM, CJ	Carácter bajo el minotauro y bajo el jugador.
JF, JC	Fila y columna del jugador.
NP	Número de pistas.
T(I)	¿Pistas utilizadas?
G1, G2, G3	Dígitos del número sugerido por el jugador.
MC, MF	Columna y fila del minotauro.

		Inicialización	
	4	GOSUB 20000	
	5	TP=1: INTERVAL ON: ON INTERVA	
		L=10 GOSUB 7	
	6	GOTO 10	
	7	TP=TP*2: RETURN	
	10	KEY OFF: SCREEN 1: CLS: COLOR	
		15,1,1: PRINT"____MINOTAURO_M	
		ASTER_MIND"	

	15	PRINT: PRINT: PRINT: PRINT "_	
		NIVEL_DE_DIFICULTAD_(1-9)";	
	20	ND=VAL(INKEY\$): IF ND<1 OR ND	
		>9 THEN 20 ELSE PRINT ND	
	22	FOR I=1088 TO 1344 STEP 64: F	
		OR J=0 TO 7: READ Q: VPOKE I+	
		J,Q: NEXT: NEXT	
	23	VPOKE 8209,49: VPOKE 8210,97:	
		VPOKE 8211,177: VPOKE 8212,	
		145: VPOKE 8213,129	
	25	FOR I=1 TO 8: PRINT: NEXT: PR	
		INT "PULSA UNA TECLA PARA EMP	
		EZAR"	
	30	IF INKEY\$="" THEN 30	
	35	INTERVAL OFF: IF TP>1000 THEN	
		TP=TP/3: GOTO 35	
	40	T=0: NP=0: FOR I=1 TO 10: T(I	
		)=0: NEXT: V=1: NS=INT(TP): N	
		=NS: D1=INT(NS/100): D2=INT((	
		NS-D1*100)/10): D3=INT(NS-D1*	
		100-D2*10)	
		Jugada nueva	
	100	CLS: PJ=6767: PM=6575	
	110	FOR I=6304 TO 6752 STEP 32: F	
		OR J=1 TO 30: GOSUB 900: K1=	
		INT(NA*13): IF K1<10 THEN VPO	
		KE I+J,K1+48: GOTO 130CONT	
	115	IF K1=10 THEN VPOKE I+J,144:	
		GOTO 130	
	120	IF K1=11 THEN VPOKE I+J,152:	
		GOTO 130	
	125	VPOKE I+J,160	
	130	NEXT: NEXT	
	140	VPOKE 6767,48: UP=0: CM=VPEEK	
		(PM): VPOKE PM,136: VPOKE	
		6319,168: CJ=48	
	150	PRINT "_VUELTA_#": V: "_NIVE	
		L_":ND: CHR\$(11):	
	160	GOSUB 1000	

Editor

```

200 C=0:Q=1
205 X#=INKEY#: C=C+1
210 IF C=7 THEN Q=1-Q: C=0
215 IF Q<>0 THEN 230
220 IF VPEEK(PJ)=255 THEN VPOKE P
    J,CJ: Q=1: GOTO 230
225 VPOKE PJ,255: Q=1
230 IF X#="" THEN 205
240 GOSUB 950
242 IF VPEEK(PJ)=255 THEN VPOKE P
    J,CJ
245 IF X#="P" OR X#="p" THEN 300
250 IF X#="O" OR X#="o" THEN 310
255 IF X#="@" OR X#="2" THEN 320
260 IF X#=";" THEN 330
265 IF X#="-" THEN 340
270 IF X#="0" THEN 350
275 IF X#="L" OR X#="1" THEN 360
280 IF X#=":" THEN 370
285 IF X#="T" OR X#="t" THEN 380
290 GOTO 205

```

Comprobar movimientos permitidos

```

300 IF VPEEK(PJ-32)=32 THEN 205
302 UP=VPEEK(PJ)-48
304 VPOKE PJ,32: PJ=PJ-32
305 GOTO 400
310 IF VPEEK(PJ-1)=32 THEN 205
312 UP=VPEEK(PJ)-48
314 VPOKE PJ,32: PJ=PJ-1
315 GOTO 400
320 IF VPEEK(PJ+1)=32 THEN 205
322 UP=VPEEK(PJ)-48
324 VPOKE PJ,32: PJ=PJ+1
325 GOTO 400
330 IF VPEEK(PJ+32)=32 THEN 205
332 UP=VPEEK(PJ)-48
334 VPOKE PJ,32: PJ=PJ+32
335 GOTO 400

```

```

340 IF VPEEK(PJ-31)=32 THEN 205
342 UP=VPEEK(PJ)-48
344 VPOKE PJ,32: PJ=PJ-31
345 GOTO 400
350 IF VPEEK(PJ-33)=32 THEN 205
352 UP=VPEEK(PJ)-48
354 VPOKE PJ,32: PJ=PJ-33
355 GOTO 400
360 IF VPEEK(PJ+31)=32 THEN 205
362 UP=VPEEK(PJ)-48
364 VPOKE PJ,32: PJ=PJ+31
365 GOTO 400
370 IF VPEEK(PJ+33)=32 THEN 205
372 UP=VPEEK(PJ)-48
374 VPOKE PJ,32: PJ=PJ+33
375 GOTO 400
380 JF=INT(PJ/32): JC=PJ-JF*32:
    GOSUB 500: IF PJ<>PM THEN 380
385 GOTO 800

```

Comprobar movimiento

```

400 CJ=VPEEK(PJ): JF=INT(PJ/32):
    JC=PJ-JF*32
402 IF (UP>9) OR (UP<0) THEN UP=0
405 K1=VPEEK(PJ): IF K1=152 THEN
    GOSUB 600
410 IF K1=144 THEN GOSUB 650
415 IF K1=160 THEN GOSUB 750
420 IF K1=168 THEN 2000
422 K1=VPEEK(PJ)-48: IF K1<0 OR K
    1>9 THEN K1=0
425 IF U+K1>=10-ND THEN GOSUB 500
430 T=T+K1: GOSUB 1000
435 IF PJ=PM THEN 800
440 GOTO 200

```

Movimiento del minotauro

```

500 VPOKE 8198,31: MF=INT(PM/32):
    MC=PM-MF*32
502 IF MC<>CJ THEN 508

```

```

O 504 IF PM>FJ THEN MF=MF-1: GOTO
    540
O 506 MF=MF+1: GOTO 540
O 508 IF MF<>FJ THEN 515
O 510 IF MC>CJ THEN MC=MC-1: GOTO
    540
O 512 MC=MC+1: GOTO 540
O 515 IF PM>FJ THEN 522
O 517 IF MC<FJ THEN MC=MC+1: GOTO
    540
O 520 MC=MC-1: GOTO 540
O 522 MF=MF-1
O 540 K=32*MF+MC: VPOKE 8198,241:IF
    VPEEK(K)=136 THEN RETURN
O 542 VPOKE PM,CB
O 544 PM=K: CB=VPEEK(PM)
O 546 VPOKE PM,136
O 550 PLAY "n2"
O 570 RETURN

```

Jugador sobre (?)

```

O 600 GOSUB 900:I=INT(NA*10): VPOKE
    PJ,I+48
O 610 CJ=I+48: RETURN

```

Jugador sobre (X)

```

O 650 VPOKE PJ,48
O 652 IF NP>=9 THEN BEEP: GOTO 670
O 654 GOSUB 900: K2=INT(NA*9+1): IF
    T(K2)=1 THEN 654
O 656 T(K2)=1: NP=NP+1: BEEP
O 658 ON K2 GOTO 660,665,670,675,
    680,685,690,695,700
O 660 PRINT: PRINT: PRINT"Suma_de_l
    os_digitos_del":PRINT"numero_
    secreto_es_";
O 662 PRINT D1+D2+D3: PRINT CHR$(11
    );: RETURN
O 665 PRINT: PRINT: PRINT"El_numero
    _secreto_es_";

```

```

O 666 IF NS/2=INT(NS/2) THEN PRINT"
    Par": GOTO 668
O 667 PRINT"Impar"
O 668 PRINT CHR$(11);: RETURN
O 670 GOSUB 900:K=INT(NA*200)+NS:J=
    K-200: IF J<0 THEN J=0
O 672 IF K>1000 THEN K=1000
O 673 PRINT: PRINT: PRINT"El_numero
    _esta_entre":J: PRINT"y":K
O 674 PRINT CHR$(11);: RETURN
O 675 PRINT: PRINT: PRINT"El_numero
    _";
O 676 IF NS/5=INT(NS/5) THEN PRINT"
    es_divisible":GOTO 678
O 677 PRINT"no_es_divisible"
O 678 PRINT"por_5": PRINT CHR$(11);
    : RETURN
O 680 L=D1: IF D2>L THEN L=D2
O 681 IF D3>L THEN D3=L
O 682 PRINT: PRINT: PRINT"El_mayor_
    digito_del_numero": PRINT"sec
    reto_es_": L ; CHR$(11);:
    RETURN
O 685 PRINT: PRINT: PRINT"El_primer
    _digito_del_numero": PRINT"se
    creto_es";
O 686 IF D1>5 THEN PRINT"Mayor_que_
    5": GOTO 689
O 687 IF D1<5 THEN PRINT"Menor_que_
    5": GOTO 689
O 688 PRINT"Igual_a_5"
O 689 PRINT CHR$(11);: RETURN
O 690 PRINT: PRINT: PRINT"La_suma_d
    e_los_digitos": PRINT"primer_o
    _y_tercero_es";
O 692 PRINT D1+D3; CHR$(11) ; :
    RETURN
O 695 PRINT: PRINT: PRINT"Uno_de_lo
    s_digitos_del": PRINT"numero_
    secreto_es";
O 696 GOSUB 900: K=INT(NA*3): IF K=
    0 AND NS>99 THEN PRINT D2: GO
    TO 699
O 698 PRINT D3
O 699 PRINT CHR$(11);: RETURN

```

```

700 PRINT: PRINT: PRINT"El produc
to de los dígitos": PRINT"que
uno son los ceros":
701 IF D1=0 AND D2=0 THEN PRINT
D3: GOTO 708
702 IF D1=0 AND D3=0 THEN PRINT
D2: GOTO 708
703 IF D1=0 THEN PRINT D2*D3: GOT
O 708
704 IF D2=0 AND D3=0 THEN PRINT
D1: GOTO 708
705 IF D2=0 THEN PRINT D1*D3:GOTO
708
706 IF D3=0 THEN PRINT D1*D2:GOTO
708
707 PRINT D1*D2*D3
708 PRINT CHR$(11);: RETURN

```

Jugador sobre (\*)

```

750 VPOKE PJ,48
752 IF T<11 THEN RETURN
754 GOSUB 900: I=INT(NA*100): IF
I>50 THEN T=T-10: PLAY"L64N10
N8N6N4": RETURN
756 IF I<10 THEN T=T+50: PLAY"L32
N10N8N6N4N10N8N6N4": RETURN
760 T=T+10: PLAY"L64N10N8N6N4":
RETURN

```

Fin de juego

```

800 CLS: IF PM=PJ THEN 820
805 FOR I=1 TO 8: PRINTTAB(2*I)"E
NHORABUENA":PRINT: NEXT:PRINT
810 PRINT"HAS VENCIDO AL MINOTAUR
O":PRINT"TU PUNTUACION ES ";T
815 PRINT"EN";V;" VUELTAS": GOTO
840
820 FOR I=1 TO 5: PRINT TAB(2*I)"
VENCIO EL MINOTAURO": PRINT:
NEXT: PRINT

```

```

825 PRINT"El número secreto era "
;NS
840 PRINT: PRINT"PULSA UNA TECLA
PARA": PRINT"JUGAR OTRA VEZ"
845 X$=INKEY$: IF X$="" THEN 845
846 RUN 5

```

Número aleatorio

```

900 NA=(9999*NA+N) MOD 5997!: NA=
NA/5997!
910 IF NA<.2 THEN N=NA*10000+1
920 RETURN

```

Mensaje en blanco

```

950 FOR I=6208 TO 6271: VPOKE I,
32: NEXT
970 RETURN

```

Puntuación

```

1000 FOR I=1 TO 21: PRINT: NEXT: P
RINT"PUNTUACION TOTAL";T: PRI
NT"ULTIMA PUNTUACION";UP:CHR$
(11);
1010 RETURN

```

Adivinar número

```

2000 PRINT: PRINT: PRINT"ADIVINA E
L NUMERO SECRETO"
2005 INPUT"Y PULSA (RETURN) ";K$
2010 I=VAL(K$): IF I=NS THEN 800
2015 G1=INT(I/100): G2=INT((I-G1*
100)/10): G3=I-G1*100-G2*10
2020 CLS: PRINT: IF D1=G1 THEN PRI
NT"El primer dígito es correc

```

```

2025 to":PRINT
IF D2=62 THEN PRINT"El_segund
a_digito_es_correcto":PRINT
2030 IF D3=63 THEN PRINT"El_tercer
_digito_es_correcto"
2035 IF D1<>61 AND D2<>62 AND D3<>
63 THEN PRINT"Ningun_digito_e
s_correcto"
2040 FOR Z=1 TO 2000: NEXT
2045 V=V+1: GOTO 100

```

#### Datos de sprites pequeños

```

10000 DATA 165,126,219,126,102,126,
82,60
10002 DATA 136,136,80,32,80,136,136
,0
10004 DATA 112,136,8,16,32,0,32,0
10006 DATA 32,168,112,32,112,168,32
,0
10008 DATA 16,56,124,254,124,56,16
,0

```

#### Instrucciones

```

20000 CLS: INPUT"QUIERES_INSTRUCCION
ES_(S/N)?" : X$: IF X$="N" OR X
$="n" THEN 20010
20001 CLS: PRINT"PARA_MOVERTE_EN_EL
_LABERINTO_PULSA_:"
20005 PRINT:PRINT"__P-__ARriba":
PRINT"__O-__IZQUIERDA": PRI
NT"__@-2__DERECHA": PRINT"__
;__ABAJ0": PRINT"__-__A
Rriba-DERECHA:
20006 PRINT"__0__ARriba-IZQUIERDA
": PRINT"__L-1__ABAJ0-IZQUIE
RDA":PRINT"__S: __ABAJ0-DER
ECHA":PRINT"__T-t__ATRAPADO"
20007 PRINT:PRINT:PRINT"SI_PISAS_:"
: PRINT: PRINT"__X__TENDRAS_
UNA_PISTA": PRINT"__?__SUMAR

```

		AS_PUNTUACION_____ALEAT	
		ORIA": PRINT"*____ALTERAR_A_T	
		U_PUNTUACION_____ALEATORIAM	
		ENTE"	
	20008	PRINT: PRINT"PULSA_UNA_TECLA_	
		PARA_SEGUIR"	
	20009	IF INKEY\$="" THEN 20009	
	20010	RETURN	

#### Tabla de claves

4	= 234	245	= 1850	352	= 1391
5	= 3379	250	= 1823	354	= 1358
6	= 403	255	= 1792	355	= 537
7	= 1046	260	= 966	360	= 1806
10	= 3886	265	= 970	362	= 1391
15	= 2866	270	= 977	364	= 1341
20	= 3352	275	= 1886	365	= 537
22	= 3640	280	= 1013	370	= 1816
23	= 2630	285	= 1914	372	= 1391
25	= 4004	290	= 594	374	= 1357
30	= 969	300	= 1772	375	= 537
35	= 3761	302	= 1391	380	= 4729
40	= 12699	304	= 1311	385	= 391
100	= 1486	305	= 537	400	= 3883
110	= 7411	310	= 1783	402	= 2141
115	= 2238	312	= 1391	405	= 2292
120	= 2239	314	= 1327	410	= 1197
125	= 822	315	= 537	415	= 1313
130	= 320	320	= 1783	420	= 1162
140	= 4359	322	= 1391	422	= 3276
150	= 2880	324	= 1326	425	= 2151
160	= 397	325	= 537	430	= 1254
200	= 734	330	= 1772	435	= 947
205	= 1325	332	= 1391	440	= 593
210	= 1798	334	= 1310	500	= 3133
215	= 1208	335	= 537	502	= 1416
220	= 3333	340	= 1806	504	= 2293
225	= 1087	342	= 1391	506	= 1297
230	= 1036	344	= 1342	508	= 1176
240	= 345	345	= 537	510	= 2275
242	= 2129	350	= 1816	512	= 1294

515	= 959	685	= 5021	900	= 3262
517	= 2293	686	= 2757	910	= 1952
520	= 1293	687	= 2755	920	= 143
522	= 803	688	= 984	950	= 1584
540	= 3393	689	= 932	970	= 143
542	= 545	690	= 5439	1000	= 6542
544	= 1590	692	= 1513	1010	= 143
546	= 561	695	= 5243	2000	= 2692
550	= 431	696	= 4285	2005	= 1653
570	= 143	698	= 266	2010	= 1877
600	= 2555	699	= 932	2015	= 5740
610	= 973	700	= 5630	2020	= 4758
650	= 469	701	= 2407	2025	= 4459
652	= 1854	702	= 2407	2030	= 4099
654	= 3233	703	= 2127	2035	= 6067
656	= 1753	704	= 2408	2040	= 1161
658	= 3150	705	= 2125	2045	= 1258
660	= 5399	706	= 2124	10000	= 1770
662	= 2064	707	= 1018	10002	= 1572
665	= 2939	708	= 932	10004	= 1340
666	= 3220	750	= 469	10006	= 1567
667	= 744	752	= 863	10008	= 1517
668	= 932	754	= 4880	20000	= 5176
670	= 4475	756	= 3610	20001	= 3468
672	= 1542	760	= 2120	20005	= 9412
673	= 3497	800	= 1228	20006	= 8483
674	= 932	805	= 3305	20007	= 16353
675	= 1669	810	= 4344	20008	= 2636
676	= 4325	815	= 1928	20009	= 1050
677	= 1792	820	= 4060	20010	= 143
678	= 1694	825	= 2714		
680	= 1767	840	= 3734	TOTAL=	429145
681	= 1244	845	= 1602		
682	= 6102	846	= 164		

# Apéndice A

## Uso del Joystick

Para usar el "joystick" en juegos que empleen el teclado necesitarás reemplazar las llamadas de USR a la posición 60000 (que lee el teclado) por una subrutina en BASIC como la siguiente. (Utiliza STICK(1) para leer del mando 1 o STICK(2) para el mando 2, o STICK(0) para leer del teclado.)

A = STICK(1)

IF (A = 0) THEN RETURN (al programa principal)

IF (A = 1) OR (A = 2) OR (A = 8) THEN (mover arriba)

IF (A = 8) OR (A = 7) OR (A = 6) THEN (mover a la izquierda)

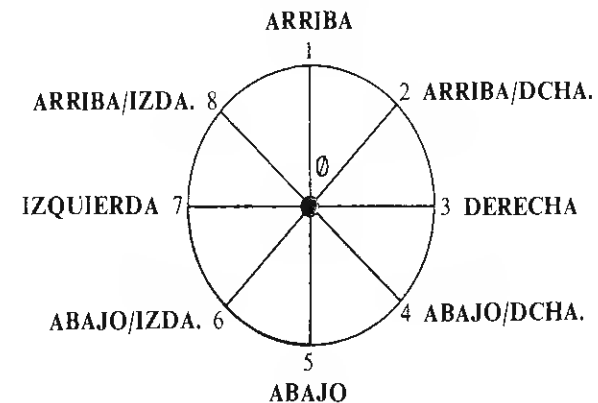
IF (A = 6) OR (A = 5) OR (A = 4) THEN (mover abajo)

IF (A = 2) OR (A = 3) OR (A = 4) THEN (mover a la derecha)

Estas sentencias de "mover" podrían ser llamadas USR a la posición 60118, acompañadas de los POKES adecuados (ver apéndice C, "Programa soporte").

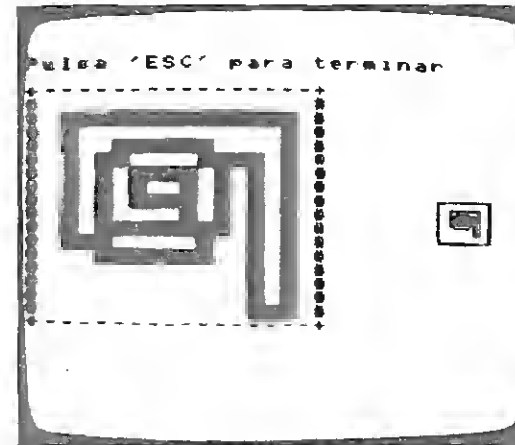
Para alterar el fuego, modifica, simplemente, cualquier referencia a STRIG(0) por STRIG(1) para el mando 1, o STRIG(2) para el mando 2.

Valores del mando, STICK(N)



# Apéndice B

## Constructor de sprites



Este programa facilita la creación de sprites moviendo el cursor y definiendo puntos en los lugares deseados. Cuando has terminado, el programa te facilita los números que necesitas para crear el sprite en tu programa. Si vas a poner estos números en una sentencia DATA, con las siguientes instrucciones crearás un sprite de  $16 \times 16$ :

```
10 A$=" "; FOR I=1 TO 32: READ Q: A$=A$+CHR$(Q): NEXT I:  
   SPRITE$(0)=A$
```

Si cambias "32" por "8" crearás un sprite de  $8 \times 8$ ; puedes utilizar este tipo de sprites para redefinir caracteres.

Las instrucciones de este programa están incluidas en él mismo, y, además, a la derecha de la matriz te aparece el sprite según lo vas construyendo.

## SUGERENCIAS AL PROGRAMA

Puedes establecer el color de sprite añadiendo el número de color al final de la sentencia "PUT SPRITE" en las líneas 120 y 520.

Si quieres ver el sprite en modo ampliado, puedes hacer que el programa dé a elegir entre los modos, y modificar las cajas de alrededor de los sprites adecuadamente.

## Variables del programa

MA(I, J) Matriz del sprite (1 a 0).  
PC Posición del cursor.  
CB Carácter bajo el cursor.  
FI Fila.  
CO Columna.

		Inicialización	
10	KEY OFF: COLOR 1,15,15: SCREE		
	N 1,2: PRINT "__CONSTRUCTOR_DE		
	_SPRITES"		
15	PRINT: PRINT: PRINT: INPUT"EL		
	IGE_8*8_('8')_D_____		
	_16*16_('16')";X\$		
20	IF X\$="16" THEN 500 ELSE IF X		
	\$="8" THEN 100 ELSE PRINT"REP		
	ITE": GOTO 15		
		Preparar pantalla (sprite 8 x 8)	
100	CLS:PRINT: PRINT: PRINT"_PuIs		
	a_'ESC'_para_finalizar."		
105	PRINT: PRINT: PRINT"____+----		
	---+":FOR I= 1 TO 8 : PRINT"_		
	_#_____": NEXT: PRINT"_		
	_+-----+"		
110	PRINT: PRINT: PRINT"___Utiliz		
	alas_tecclas_del____cursor_pa		
	ra_mover": PRINT: PRINT"_PuI		
	sa_'P'_para_fijar_un_____pun		

		to.": PRINT : PRINT"_PuIsa_'	
		B'_para_borrar."	
115	PC=6473-64: CB=32: FI=4:CO=4		
120	A\$="": SPRITE\$(0)=A\$: PUT SPR		
	ITE 0, (167,86)		
125	VPOKE 6484,24: VPOKE 6485,23:		
	VPOKE 6486,25: VPOKE 6516,22:		
	VPOKE 6518,22: VPOKE 6549,23:		
	VPOKE 6548,26: VPOKE 6550,27		
		Control (sprite 8 x 8)	
200	X\$=INKEY\$: VPOKE PC,255		
250	IF X\$="" THEN 200		
255	IF X\$=CHR\$(27) THEN 400		
260	IF X\$="P" OR X\$="p" THEN VPOK		
	E PC,CB: GOTO 300		
265	IF X\$="B" OR X\$="b" THEN VPOK		
	E PC,CB: GOTO 310		
270	IF X\$=CHR\$(30) THEN 320		
275	IF X\$=CHR\$(29) THEN 330		
280	IF X\$=CHR\$(31) THEN 340		
285	IF X\$=CHR\$(28) THEN 350		
290	GOTO 200		
		Mover/dibujar/borrar	
300	IF VPEEK(PC)=255 THEN 200 ELS		
	E CB=255: MA(FI,CO)=1: GOTO		
	380		
310	IF VPEEK(PC)=32 THEN 200 ELSE		
	CB=32: MA(FI,CO)=0: GOTO 380		
320	IF FI<2 THEN 200 ELSE FI=FI-1		
	: VPOKE PC,CB: CB=VPEEK(PC-32		
	): PC=PC-32: GOTO 200		
330	IF CO<2 THEN 200 ELSE CO=CO-1		
	: VPOKE PC,CB: CB=VPEEK(PC-1)		
	: PC=PC-1: GOTO 200		
340	IF FI>7 THEN 200 ELSE FI=FI+1		
	: VPOKE PC,CB: CB=VPEEK(PC+32		
	): PC=PC+32: GOTO 200		



```

350 IF CO>7 THEN 200 ELSE CO=CO+1
: VPOKE PC,CB: CB=VPEEK(PC+1)
: PC=PC+1: GOTO 200

```

Visualizar sprite 8 x 8

```

380 K2=14335+FI:K=VPEEK(K2): K1=2
^ (8-CO): IF MA(FI,CO)=1 THEN
K=K+K1: VPOKE K2,K: GOTO 200
385 K=K-K1: VPOKE K2,K: GOTO 200

```

Fin (sprite 8 x 8)

```

400 CLS: PRINT "Numeros para las s
entencias DATA": PRINT: PRI
NT: PRINT
410 FOR I=14336 TO 14342: PRINT U
SING"###"; VPEEK(I): PRINT "\
";: NEXT
415 PRINT: PRINT: PRINT: PRINT US
ING"###"; VPEEK(14343)
420 FOR I=1 TO 4: PRINT: NEXT: PR
INT "Pulsa una tecla para empe
zar de nuevo": PUT SPRITE 0, (
127,150)
430 IF INKEY$="" THEN 430 ELSE
RUN

```

Preparar pantalla (sprite 16 x 16)

```

500 CLS: PRINT "Pulsa 'ESC' para t
erminar": PRINT
505 PRINT "+-----+": FO
R I=1 TO 16: PRINT "#
#####
#####":NEXT: PRINT "+-----
-----+"
510 DIM MA(16,16): PC=6474: CB=32
: FI=8: CO=8
520 A$="": SPRITE$(0)=A$: PUT SPR
ITE 0, (216,86)

```

```

525 VPOKE 6490,24: VPOKE 6491,23:
VPOKE 6492,23
526 VPOKE 6493,25: VPOKE 6522,22:
VPOKE 6525,22
527 VPOKE 6554,22: VPOKE 6557,22:
VPOKE 6587,23
528 VPOKE 6588,23: VPOKE 6586,26:
VPOKE 6589,27

```

Control (sprite 16 x 16)

```

600 X$=INKEY$: VPOKE PC,255
650 IF X$="" THEN 600
655 IF X$=CHR$(27) THEN 800
660 IF X$="P" OR X$="p" THEN VPOK
E PC,CB: GOTO 700
665 IF X$="B" OR X$="b" THEN VPOK
E PC,CB: GOTO 710
670 IF X$=CHR$(30) THEN 720
675 IF X$=CHR$(29) THEN 730
680 IF X$=CHR$(31) THEN 740
685 IF X$=CHR$(28) THEN 750
690 GOTO 600

```

Mover/dibujar/borrar

```

700 IF VPEEK(PC)=255 THEN 600 ELS
E CB= 255: MA(FI,CO)=1 : GOTO
780
710 IF VPEEK(PC)=32 THEN 600 ELSE
CB=32: MA(FI,CO)=0: GOTO 780
720 IF FI<2 THEN 600 ELSE FI=FI-1
: VPOKE PC,CB: CB=VPEEK(PC-32
): PC=PC-32: GOTO 600
730 IF CO<2 THEN 600 ELSE CO=CO-1
: VPOKE PC,CB: CB=VPEEK(PC-1)
: PC=PC-1: GOTO 600
740 IF FI>15 THEN 600 ELSE FI=FI+
1: VPOKE PC,CB: CB=VPEEK(PC+
32): PC=PC+32: GOTO 600
750 IF CO>15 THEN 600 ELSE CO=CO+
1: VPOKE PC,CB: CB=VPEEK(PC+1
): PC=PC+1: GOTO 600

```

		Visualizar sprite 16 x 16	
780	IF CO<9 THEN K1=0 ELSE K1=16		
785	K2=14335+FI+K1: K=VPEEK(K2):		
	IF CO<9 THEN K1=2^(9-CO) ELSE		
	K1=2^(16-CO)		
790	IF MA(FI,CO)=1 THEN K=K+K1:		
	VPOKE K2,K: GOTO 600		
795	K=K-K1:VPOKE K2,K: GOTO 600		
		Fin (sprite 16 x 16)	
800	CLS: PRINT"Numeros_para_las_s		
	entencias_'DATA': PRINT"_Lee		
	r_de_izquierda_a_derecha": PR		
	INT		
810	FOR I=14335 TO 14356 STEP 7:		
	PRINT: PRINT: FOR J=1 TO 7 :		
	PRINT USING"###": VPEEK(I+J):		
	: PRINT "_": NEXT: NEXT		
815	PRINT: PRINT: FOR I=14364 TO		
	14367: PRINT USING"###": VPEE		
	K(I): PRINT"_": NEXT		
820	FOR I=1 TO 3: PRINT: NEXT: PR		
	INT"_Pulsa_una_tecla_para_emp		
	ezar_de_nuevo": PUT SPRITE 0,		
	(120,165)		
830	IF INKEY\$="" THEN 830ELSE RUN		

350	= 6170	528	= 1780	740	= 5978
380	= 7434	600	= 1340	750	= 6005
385	= 2007	650	= 920	780	= 1845
400	= 4714	655	= 1292	785	= 6417
410	= 2988	660	= 3076	790	= 3206
415	= 1934	665	= 3078	795	= 1879
420	= 6590	670	= 1516	800	= 7629
430	= 1484	675	= 1516	810	= 5442
500	= 3181	680	= 1532	815	= 3392
505	= 5351	685	= 1532	820	= 6635
510	= 2712	690	= 483	830	= 1374
520	= 2297	700	= 4271		
525	= 1488	710	= 3751	TOTAL=	225186
526	= 1556	720	= 5902		
527	= 1704	730	= 5935		

## Tabla de claves

10	= 3770	125	= 5134	280	= 1389
15	= 3548	200	= 1340	285	= 1389
20	= 4278	250	= 1032	290	= 593
100	= 3593	255	= 1449	300	= 4463
105	= 4510	260	= 2948	310	= 3939
110	= 13738	265	= 2916	320	= 6140
115	= 2367	270	= 1373	330	= 6173
120	= 2258	275	= 1373	340	= 6137

# Apéndice C

## Programa soporte

Este programa (se necesita en algunos programas de este libro) debe ser cargado y ejecutado antes de que cargues tu programa.

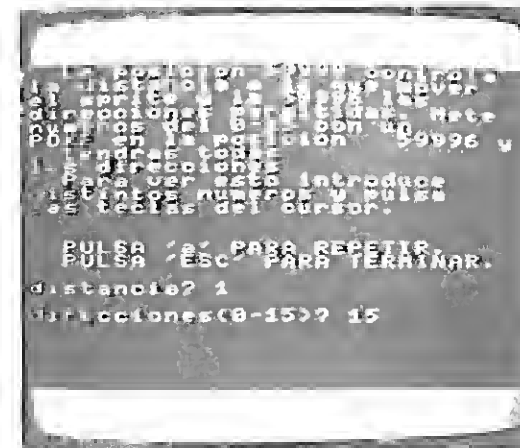
Pasos a seguir:

1. Mete este programa, utiliza el verificador para depurarlo y después grábalo como "SOPORTE".
2. Cárgalo en memoria y ejecútalo.
3. Carga en memoria tu programa depurado y ejecutado; él llamará y usará a éste cuando sea necesario.

○	10	CLS: PRINT "SOPORTE"	○
	20	FOR I=60000! TO 60217!: READ	
○		Q: POKE I,Q : NEXT	○
	1000	DATA 219,170,230,240,33,92,	
		234,94,246,8,0,0,0,0,0,0,	○
○		0,0,0,211,170,219,169,71,0,0,	○
		0,0,0,0,0,0	
	1010	DATA 0,0,0,230,128,194,151,	
○		234,203,91,202,151,234,62,3,	○
		50,94,234,195,209,234,120,230	
		,64,194,170,234,203,83,202,	
○		170,234,62,2	○
	1020	DATA 50,94,234,195,209,234,	
		120,230,32,194,189,234,203,67	○
○		,202,189,234,62,0,50,94,234,	○
		195,209,234,120,230,16,194,	
		208,234,203,75,202,208,234,62	○
○		,1,50,94,234,195,209,234,201,	○
		62,0,50,93,234,58	
	1030	DATA 93,234,203,39,203,39,79,	
○		6,0,33,0,27,9,58,95,234,71,58	○
		,94,234,87,254,0,194,250,234,	
○		205,27,235,121,152,79,195,44,	○
		235,254	

1040	DATA 1, 194, 6, 235, 35, 205, 27, 235, 195, 244, 234, 254, 2, 194, 20, 235, 205, 27, 235, 121, 128, 79, 195 , 44, 235, 35, 205, 27, 235, 195, 14, 235, 125, 243, 211
1050	DATA 153, 124, 211, 153, 0, 0, 0, 0, 219, 152, 79, 183, 251, 201, 125, 243, 211, 153, 124, 246, 64, 211, 153, 121, 211, 152, 251, 201

10	=	1044	1010	=	10317	1040	=	11087
20	=	2233	1020	=	21367	1050	=	8282
1000	=	7068	1030	=	10298	TOTAL=		71696



## SUGERENCIAS A ESTE PROGRAMA

La segunda parte sólo permite al usuario elegir direcciones para mover el sprite, pero puedes modificar el "1" de la línea 90 para variar la velocidad, o puedes también reestructurar el programa para que dé a escoger la velocidad al ejecutarse el programa.

### Variables del programa

X\$ Tecla apretada.  
V Temporal.

0		Inicialización	0
6	REM MAQUINA SOPORTE		0
7	REM DEL PROGRAMA		0
8	REM VER APENDICES		0
10	COLOR 15,4,7: SCREEN 1,1: KEY OFF		0
15	FOR I=1 TO 8: READ Q: A\$=A\$+CHR\$(Q): NEXT: SPRITE\$(0)=A\$: GOTO 200		0
0		Primera parte: leer del teclado	0
20	CLS: PRINT TAB(3)"DEMOSTRACION DEL CODIGO": PRINT TAB(5)"MAQUINA SOPORTE DEL": PRINT TAB(10)"PROGRAMA": PRINT: PRINT: PRINT: PRINT TAB(11)"PARTE 1": PRINT: PRINT		0
25	DEFUSR0=60000!:PRINT"La primera rutina empieza en la posicion 60000. Sirve para mover el sprite con las teclas del cursor.":PRINT"Las 4 direcciones posibles estan numeradas como sigue.":PRINT		0
26	PRINT:PRINT"PULSA UNA TECLA"		0

27	IF INKEY\$="" THEN 27	0	
30	PRINT: PRINT TAB(13)"0": PRINT: PRINT TAB(13)"U": PRINT: PRINT TAB(13)"V": PRINT: PRINT TAB(9)"1_UWWW UL3": PRINT TAB(13)"V": PRINT: PRINT TAB(13)"U": PRINT: PRINT: PRINT TAB(13)"2": PRINT: PRINT"PULSA UNA TECLA PARA SEGUIR"	0	
40	IF INKEY\$="" THEN 40	0	
45	CLS:PRINT"La posicion 59999 controla la distancia a la que mover el sprite y la 59996 las direcciones permitidas. Mete numeros del 0-15 con un POKe en la posicion 59996 y tendras todas"	0	
46	PRINT"las direcciones"	0	
47	PRINT "Para ver esto introduce distintos numeros y pulsa las teclas del cursor.": PRINT: PRINT	0	
50	PRINT "Para pulsar 'a' PARA REPETIR.":PRINT"Para pulsar 'ESC' PARA TERMINAR.": PRINT	0	
55	INPUT "distancia";D: PRINT: INPUT "direcciones(0-15)";X\$	0	
60	PRINT: PRINT: PUT SPRITE 0,(120,175),1	0	
65	POKE 59999!,D: POKE 59996!,VAL(X\$): I=USR0(I): A\$=INKEY\$: IF A\$="A" OR A\$="a" THEN 50 ELSE IF A\$=CHR\$(27) THEN PUT SPRITE 0,(100,200): GOTO 200 ELSE 65	0	
<div style="border: 1px solid black; padding: 10px; margin: 10px auto; width: fit-content;">Segunda parte: movimiento de sprites</div>			0
75	CLS: PRINT TAB(11)"PARTE 2": PRINT: PRINT"Esta rutina empieza en la pos. 60118 y mueve el sprite que debe situarse en la pos. 59997, la direcc	0	

```

ion_en_Ia_59998_y_Ia_dista
ncia_en_Ia_59999."
80 PRINT:PRINT "Pulsa RETURN
para cambiar la direccion.":
PRINT:PRINT "Pulsa ESC para
terminar.":PRINT
85 DEFUSR1=60118!:POKE 59997!,0
:PUT SPRITE 0,(10,175),1
90 POKE 59998!,1:POKE 59999!,1
95 I=USR1(I):X$=INKEY$:IF X$=
"" THEN 95 ELSE IF X$=CHR$(27
) THEN PUT SPRITE 0,(100,200)
:GOTO 200 ELSE IF X$ <> CHR$(
13) THEN 95
96 INPUT "Direccion(0-3)";X$:K=V
AL(X$):IF K<0 OR K>3 THEN PR
INT "Repite":GOTO 96 ELSE PO
KE 59998!,K:CLS:PRINT "Pul
sa RETURN para cambiar de d
ireccion.":PRINT "Pulsa ESC
para terminar.":GOTO 95

Menú

200 CLS:PRINT "DEMOSTRACION DEL CODIGO
O MAQUINA"
205 PRINT:PRINT TAB(2) "Pulsa:":
PRINT:PRINT TAB(3) "1 para
demostrar parte 1":PRINT:PRIN
T TAB(3) "2 para demostrar par
te 2":PRINT:PRINT TAB(3) "F
para finalizar"
210 PRINT:PRINT:PRINT TAB(2) "seg
un la opcion que elijas"
220 X$=INKEY$:IF X$="1" THEN 20
ELSE IF X$="2" THEN 75 ELSE I
F X$="F" OR X$="f" THEN SCREE
N 1:PRINT:PRINT:PRINT TAB(
10) "---ADIOS!!":END ELSE 220
1000 DATA 24,60,102,255,255,24,36,
66

```

## Tabla de claves

5	= 0	40	= 979	90	= 1523
6	= 0	45	= 27897	95	= 8437
7	= 0	46	= 1821	96	= 19734
8	= 0	47	= 9224	200	= 2588
10	= 1235	50	= 5390	205	= 11105
15	= 4280	55	= 3738	210	= 3758
20	= 10588	60	= 1400	220	= 8796
25	= 25606	65	= 10795	1000	= 1583
26	= 1635	75	= 22507		
27	= 966	80	= 8371	TOTAL=	207884
30	= 11049	85	= 2879		

# Apéndice D

## Rutina código máquina de movimiento de pantalla



Se usan varias subrutinas en código máquina para mover la pantalla en este libro, pero son todas muy parecidas a la del siguiente ejemplo:

○	5	REM EJECUTA EL CODIGO	○
	6	REM MAQUINA SOPORTE	
○	7	REM DEL PROGRAMA	○
	8	REM VER APENDICES	
	10	SCREEN 1: COLOR 15, 4, 7 :KEY	
○		OFF: FOR I=60350! TO 60401!:	○
		READ Q: POKE I,Q : NEXT	
	15	DEFUSR=60350!: POKE 60346!,0:	
○		POKE 60347!,24	○
	16	POKE 60348!,255: POKE 60349!,	
		26	
○	20	PRINT "___DEMOSTRACION_DE_LA_	○
		RUTINA":PRINT"_____EN_CODIG	
○		O_MAQUINA":PRINT "_____QUE_AN	○
		IMA_LA_PANTALLA"	

○	25	FOR I=1 TO 10: PRINT: NEXT: P	○
○		RINT" PULSA UNA TECLA PARA EM	○
○		PEZAR":PRINT : PRINT" PULSA	○
○	30	IF INKEY#="" THEN 30	○
○	40	CLS: PRINT" Esta rutina de an	○
○		imacion de pantalla es un eje	○
○		mplo de las": PRINT" usadas e	○
○		n este libro."	○
○	45	FOR I=1 TO 5:PRINT:NEXT: PRIN	○
○		T " PULSA 'RETURN':PRI	○
○		NT:PRINT" CUANDO ESTES LI	○
○		STO"	○
○	50	IF INKEY#=CHR\$(13) THEN 55	○
○		ELSE 50	○
○	55	D=USR(D): X#=INKEY#: IF X#=CH	○
○		R\$(27) THEN SCREEN 1: PRINT "	○
○		----ADIOS!!!!": END	○
○	60	GOTO 55	○
○	10000	DATA 6,31,42,186,235,43,35,	○
○		205,27,235,120,254,31,202,215	○
○		,235,4,43,205,44,235,35,195,	○
○		227,235,6,0,17,31,0,25,205,44	○
○		,235,183,237,82,237,91,188,	○
○		235,123,189,194,196,235,122,	○
○		188	○
○	10010	DATA 194,196,235,201	○

## Tabla de claves

5	= 0	16	= 1614	50	= 1782
6	= 0	20	= 13948	55	= 5270
7	= 0	25	= 10377	60	= 444
8	= 0	30	= 969	10000	= 18692
10	= 3647	40	= 14002	10010	= 962
15	= 2295	45	= 9904	TOTAL=	83906



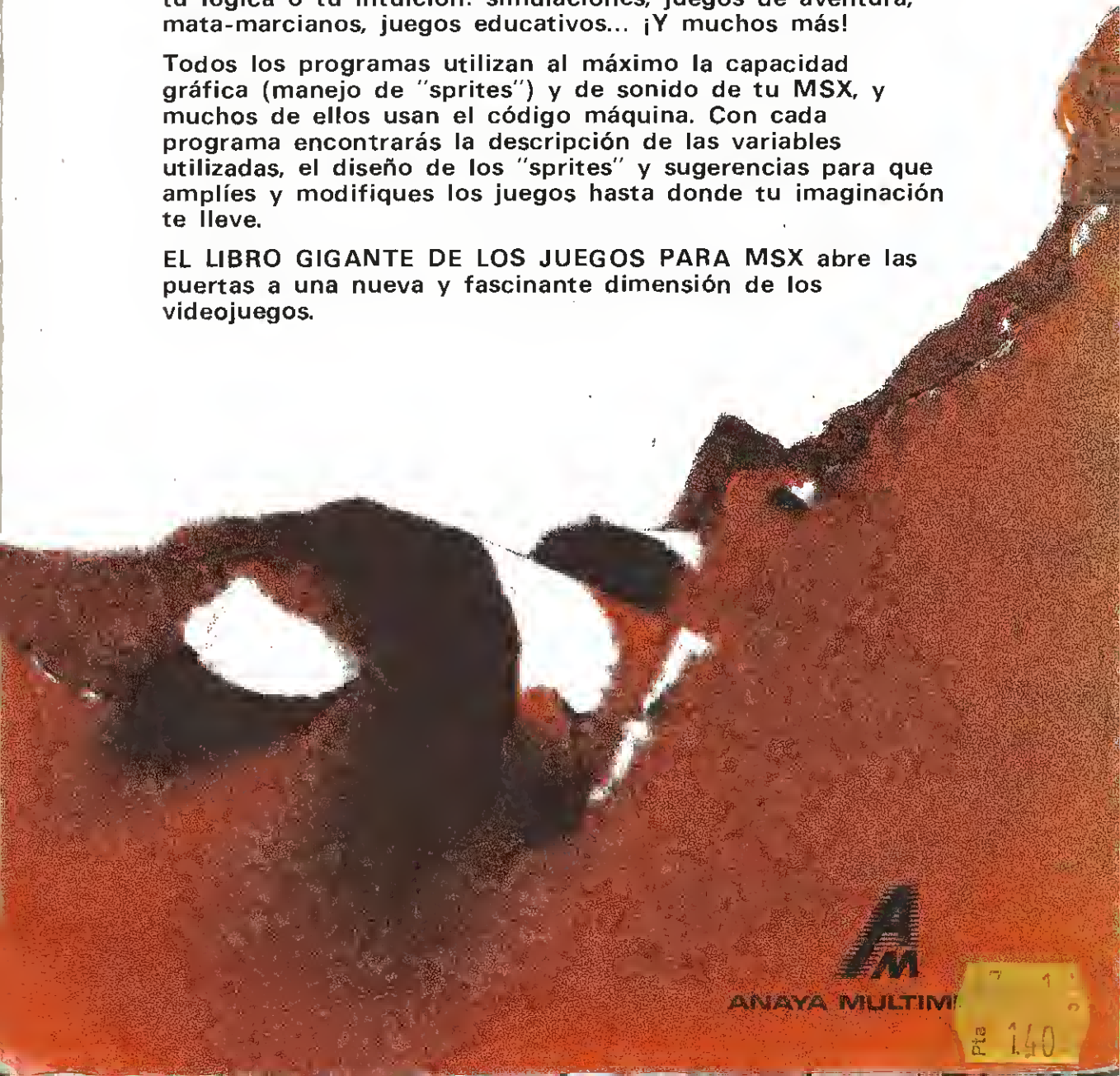
¿Está tu MSX preparado para la invasión de las cucarachas cósmicas, de los espías del planeta Circón y de las naves-robot suicidas? ¿Conoces al gorila de Godofredo?

**EL LIBRO GIGANTE DE LOS JUEGOS PARA MSX** es la recopilación más fascinante que puedes encontrar de juegos que desafían a tu microordenador MSX.

Los listados están especialmente estructurados para que su transcripción y depuración sean fáciles y rápidas. Encontrarás juegos que son todo un reto para tus reflejos, tu lógica o tu intuición: simulaciones, juegos de aventura, mata-marcianos, juegos educativos... ¡Y muchos más!

Todos los programas utilizan al máximo la capacidad gráfica (manejo de "sprites") y de sonido de tu MSX, y muchos de ellos usan el código máquina. Con cada programa encontrarás la descripción de las variables utilizadas, el diseño de los "sprites" y sugerencias para que amplíes y modifiques los juegos hasta donde tu imaginación te lleve.

**EL LIBRO GIGANTE DE LOS JUEGOS PARA MSX** abre las puertas a una nueva y fascinante dimensión de los videojuegos.



ANAYA MULTIMEDIA

Pla 140